

NO-A189 484

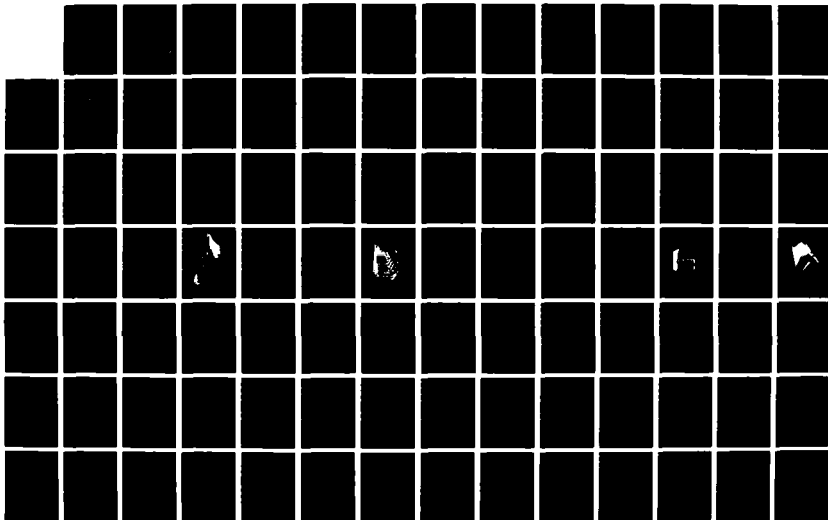
A SIMULATION OF THE DEFENSE LOGISTICS AGENCY'S  
INTEGRATED MATERIALS COMPLEX(U) AIR FORCE INST OF TECH  
WRIGHT-PATTERSON AFB OH SCHOOL OF ENGINEERING  
R B BERRY DEC 87 AFIT/GOR/ENS/87D-2

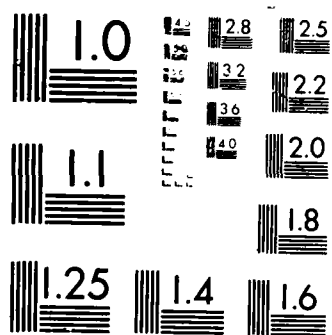
1/2

UNCLASSIFIED

F/G 15/5

NL





DTIC FILE COPY

①

AD-A189 484

AFIT/GOR/ENS/87D-2

A SIMULATION OF THE  
DEFENSE LOGISTICS AGENCY'S  
INTEGRATED MATERIALS COMPLEX  
THESIS

Richard B. Berry  
Captain, USAF  
AFIT/GOR/ENS/87D-2

DTIC  
ELECTE  
MAR 02 1988  
S H D

Approved for public release; distribution unlimited

88 3 01 088

AFIT/GOR/ENS/87D-2

A SIMULATION OF THE  
DEFENSE LOGISTICS AGENCY'S  
INTEGRATED MATERIALS COMPLEX

THESIS

Presented to the Faculty of the School of Engineering  
of the Air Force Institute of Technology  
Air University  
In Partial Fulfillment of the  
Requirements for the Degree of  
Master of Science in Operations Research

Richard B. Berry, B.A.  
Captain, USAF

December 1987

Approved for public release; distribution unlimited

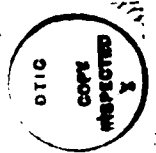
## PREFACE

The purpose of this thesis was to develop a simulation model to evaluate the effect of automated guided vehicles (AGVs) and automated storage and retrieval (ASR) system on the performance of the Defense Logistics Agency's Integrated Materials Complex (IMC). The IMC is an automated warehouse facility.

The simulation was tested extensively by varying the number of AGVs and the method in which the ASR system operates. However, the amount of testing performed was limited by the amount of available computer time. Although the simulation was written for a specific facility, it can be easily adapted to similar facilities.

While working on this thesis, I have received a great deal of help from others. I deeply indebted to my faculty advisor, Maj Joseph R. Litko for his helpful insights and assistance in time of need. I also wish to thank my readers, Lt Col James M. Robinson and Dr Dan Reynolds. Finally, I wish to thank my wife Louise for her love and understanding that has gotten me through many long nights at the computer terminal.

- Richard B. Berry



Accession For	
NTIS GRA&I	<input checked="checked" type="checkbox"/>
DTIC TAB	<input type="checkbox"/>
Unannounced	<input type="checkbox"/>
Justification	
By	
Distribution/	
Availability Codes	
Avail and/or	
Dist	Special
A-1	

## Table of Contents

	Page
Preface . . . . .	ii
List of Figures . . . . .	vi
List of Tables . . . . .	viii
Abstract . . . . .	ix
 I. Introduction . . . . .	 1
Background . . . . .	1
Problem . . . . .	6
Objectives . . . . .	6
Scope . . . . .	7
Overview of Remaining Chapters . . . . .	7
 II. Literature Review . . . . .	 9
Introduction . . . . .	9
Computer Simulations . . . . .	9
Examples of Computer Simulation . . . . .	11
Advantages and Disadvantages of Computer Simulation . . . . .	16
Methodology . . . . .	17
Summary . . . . .	18
 III. Warehouse Operations . . . . .	 19
Introduction . . . . .	19
Performance Goal of the IMC . . . . .	19
Module and Pallet Description . . . . .	27
AGVs . . . . .	28
Description . . . . .	28
Pickup and Delivery . . . . .	28
Guidewire Paths . . . . .	29
AGV Control and Anti- Collision Measures . . . . .	29
Parking Loops . . . . .	32
Receiving . . . . .	32
Module Loading Area . . . . .	34
Module Storage Area . . . . .	35
Automated Storage and Retrieval System . . . . .	35
Description . . . . .	35
Picking Priorities . . . . .	37
Operation . . . . .	38

Consolidation . . . . .	40
Pallet Movement . . . . .	42
Summary . . . . .	44
IV. Program Description . . . . .	45
Introduction . . . . .	45
Methodology . . . . .	45
Description . . . . .	47
Defining the AGVs . . . . .	47
Track Layout . . . . .	50
Pickup and Delivery Stands . . . . .	54
AGV Movement . . . . .	54
Receiving . . . . .	56
Module Storage Area . . . . .	58
Highrise Rack and Bin Storage Area . . . . .	59
Consolidation Area . . . . .	62
Pallets . . . . .	63
Tandem and AGV Computers . . . . .	64
Events Not Simulated . . . . .	68
Summary . . . . .	68
V. Results . . . . .	69
Introduction . . . . .	69
Consolidation Pick Processing . . . . .	70
Performance Goal . . . . .	70
Effects of AGVs and Pick Mode . . . . .	71
Consolidation Workstation Workload . . . . .	72
AGV Efficiency . . . . .	73
Measuring Efficiency . . . . .	73
Effect of Pick Mode . . . . .	79
AGV Routing . . . . .	79
Waiting Time for an AGV . . . . .	84
Module Storage Area . . . . .	87
Commentary on SLAM . . . . .	87
Summary . . . . .	91
VI. Conclusions and Recommendations . . . . .	92
Conclusions . . . . .	92
Recommendations . . . . .	96
Appendix A: SLAM Simulation Code . . . . .	A-1
Appendix B: Additional Fortran Code . . . . .	B-1
Appendix C: Parameter File . . . . .	C-1
Appendix D: AGV Pathway Figures . . . . .	D-1

Bibliography . . . . .	BIB-1
Vita . . . . .	BIB-3



## List of Figures

Figure	Page
1. Sample AGV . . . . .	2
2. Sample Storage and Retrieval Machine . . . . .	3
3. Layout of the Receiving Building . . . . .	20
4. Layout of the Highrise Rack and Bin Storage Building . . . . .	21
5. Layout of the Consolidation Building . . . . .	22
6. Layout of the Pallet Storage Building . . . . .	23
7. IMC Layout . . . . .	24
8. Flow of Modules in the IMC . . . . .	25
9. Flow of Pallets in the IMC . . . . .	26
10. Pathway Example . . . . .	29
11. Control Point Example #1 . . . . .	31
12. Control Point Example #2 . . . . .	31
13. Receiving Area . . . . .	33
14. Highrise Rack and Bin Storage Area . . . . .	36
15. ASR Machine Example . . . . .	39
16. Consolidation Area . . . . .	41
17. Pallet Storage Area . . . . .	43
18. AGV and Buffer Distance . . . . .	48
19. Checkzone Example . . . . .	49
20. Intersection Control Point . . . . .	50
21. Endpoint Control Point . . . . .	51
22. IMC Segment . . . . .	52
23. Simulation Segment . . . . .	53

24.	Order Processor . . . . .	65
25.	Flow of Orders . . . . .	67
26.	Interval AGV Fleet Efficiency - Pick Mode A . . . . .	74
27.	Interval Fleet Efficiency - Pick Mode B . . . . .	75
28.	Cumulative AGV Fleet Efficiency - Pick Mode A . . . . .	77
29.	Cumulative AGV Fleet Efficiency - Pick Mode B . . . . .	78
30.	Interval AGV Fleet Efficiency - 25 AGVs . . . . .	80
31.	Cumulative AGV Fleet Efficiency - 25 AGVs . . . . .	81
32.	Number of Modules Awaiting Transportation - Pick Mode A . . . . .	85
33.	Number of Modules Awaiting Transportation - Pick Mode B . . . . .	86
34.	Module Storage Area Occupancy Rate - Pick Mode A . . . . .	88
35.	Module Storage Area Occupancy Rate - Pick Mode B . . . . .	89
36.	Control Points - Module Loading Area . . . . .	D-1
37.	Control Points - Module Storage Area . . . . .	D-2
38.	Control Points - 3 P&M . . . . .	D-3
39.	Control Points - LTL . . . . .	D-4
40.	Control Points - South End of Highrise . . . . .	D-5
41.	Control Points - Middle Highrise . . . . .	D-6
42.	Control Points - North End of Highrise . . . . .	D-7
43.	Control Points - Consolidation . . . . .	D-8
44.	Control Points - Pallet Storage . . . . .	D-9

## *List of Tables*

Table	Page
1. Module Attributes . . . . .	55
2. Order Types . . . . .	65
3. Simulation Run Descriptions . . . . .	69
4. Total Picks Processed by the Consolidation Workstations . . . . .	71
5. Results of Variance Test on Effects of AGV Fleet Size and Pick Mode . . . . .	72
6. Consolidation Workstation Workload . . . . .	73
7. Vehicle Delivery Routes . . . . .	82
8. Number of AGV Trips Over a Given Route. . . . .	82
9. Average Travel Time Along a Given Route . . . . .	83
10. Results of Variance Test on Route Trips . . . . .	83
11. Results of Variance Test on Trip Times. . . . .	84

*Abstract*

The purpose of this research was to show the effects the number of AGVs, and number of pickup and delivery stands, and the storage and retrieval machines have on the performance of the Defense Logistics Agency's Integrated Material Complex (IMC). The research had three basic objectives: First, design a model that would simulate the effects of the number of AGVs in the IMC, simulate the storage and retrieval machines to evaluate their performance, and determine if the number of pickup/delivery stands in the IMC is sufficient. Second, evaluate the results of the simulation with regards to the performance goal of the IMC. Third, make recommendations, based on findings, to the DLA concerning the effectiveness of proposed or implemented changes in the IMC.

A computer simulation model was designed using the SLAM simulation language. Four different vehicle fleet sizes and two automated storage and retrieval machine picking modes were simulated. The performance measure of the IMC is to take 15,500 items from the highrise rack and bin and storage area, transport them by AGV to the consolidation area, and process for shipping, all in an eight hour shift. The results showed that the performance goal was always met

regardless of the AGV fleet size (20 to 35 vehicles) or pick method used. Analysis of the data confirmed that the AGV fleet size and pick methods had no apparent effect on achieving the performance goal.

However, if the performance goal was increased, a smaller fleet operating near full efficiency may not be able to move all the items from storage to the consolidation area. Thus, the size of the vehicle fleet would start to affect the performance of the IMC. A higher performance goal may also cause the pick mode to have more of an effect on the operation of the IMC.

A SIMULATION OF THE  
DEFENSE LOGISTIC AGENCY'S  
INTEGRATED MATERIAL COMPLEX

*1. Introduction*

*Background*

The Defense Logistic Agency (DLA) provides worldwide supply and logistic support to the United States military services. Nearly 400,000 military customers depend on the DLA, which ships more than 17 million items each year through a worldwide complex of depots, storage facilities, and fuel terminals. To serve its customers in the New England area and Europe, the DLA has a major supply depot in Mechanicsburg, PA. This depot is the site of an automated warehouse called the Integrated Material Complex (IMC), which is currently under construction.

The warehouse is designed to use the latest materials handling technology, such as automated guided vehicles (AGV), storage and retrieval machines, automated conveyor belts, and rotary storage racks. The AGVs are driverless vehicles used to transport pallets and modules (boxes) of parts to various work stations throughout the warehouse (fig. 1). Storage and retrieval machines are used to stock and remove items from aisles of bins and racks (fig. 2). Automated conveyor belts are used to transport items placed

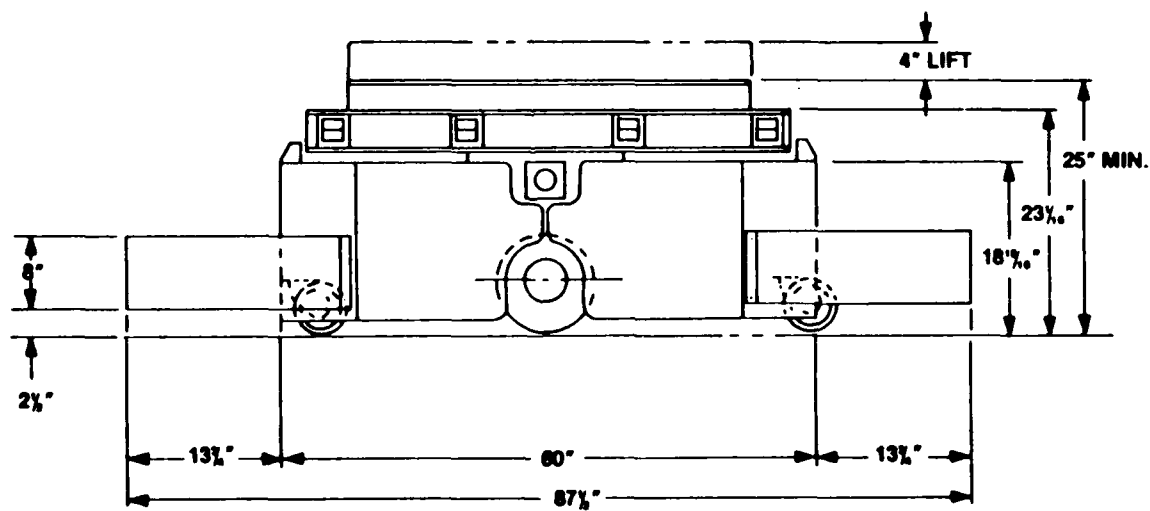
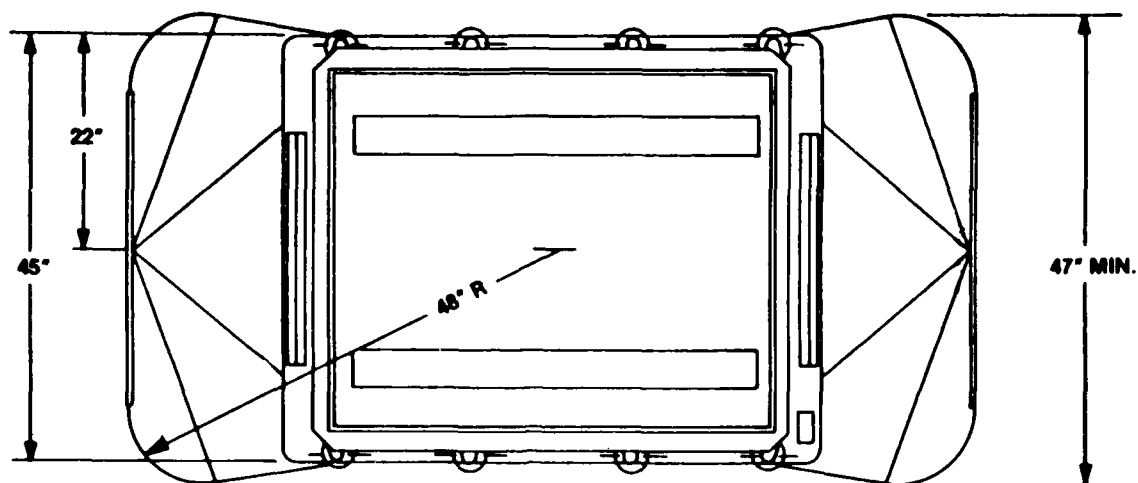


Figure 1. Sample AGV

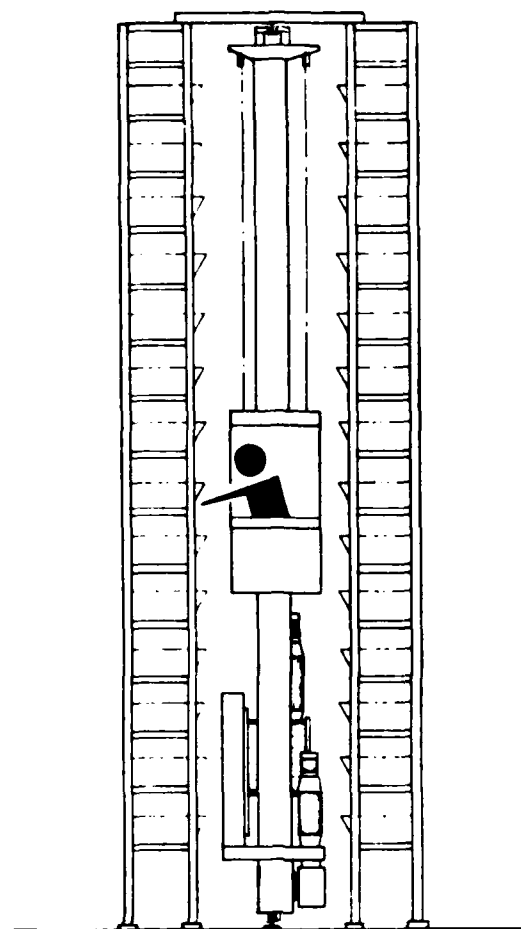


Figure 2. Sample Storage and Retrieval Machine

in totes (small plastic bins) to various locations within a building. Rotary storage racks or "carousels" are computer controlled racks used to consolidate multi-item orders.

The IMC is the DLA's largest attempt to automate an entire warehouse facility. Older DLA facilities are much more dependent on slower non-automated systems. An automated facility allows more to be done with less people, space, and money.

Fewer people are needed to run the machinery, resulting in lower operating costs. In the storage areas, the rows are made narrower because the storage and retrieval machines



need less operating space. Narrower aisles allow more storage space to be used in a defined area. Other advantages include faster processing of orders and lower operating costs. Although, the initial investment in equipment will be substantially higher than a non-automated facility, the reduced operating costs over the life of the automated facility still make it the best choice. It should be noted that this thesis does not attempt to validate any cost savings, but rather focuses on the operation of the system.

There are still some potential disadvantages to the DLA's plan for the IMC. The IMC is still under construction and is not projected to be finished until January, 1989. Until the facility is finished, there will be no way to determine its actual performance. Since this facility is the first of its kind, the DLA does not have another similar facility to compare the IMC to. However, performance of individual components of the facility can be estimated from historical or manufacturers' data. Changes made in the design to save costs or make changes in performance must be evaluated for their overall impact on the facility.

The performance goal of the IMC is to move 15,500 line items per day from the bin and rack storage area to the consolidation work area by way of modules transported by AGVs (3:2-10). This goal was determined to be sufficient to meet the needs of the customers serviced by the IMC. Yet, this goal was not met in a simulation of the facility.

A contractor, AutoSimulation Inc., developed a simulation model to estimate the actual performance of the IMC. The results of the simulation showed the warehouse would not meet the goal of moving 15,500 line items per day to the consolidation work area. The study concluded that the problem was a result of the way the AGVs were programmed. The AGVs were not moving the pick modules from the highrise to the consolidation area in the most efficient manner. The AGVs kept taking the pick modules to the same small number of the 32 consolidation workstations. This caused the consolidation workload to backup and prevented them from processing 15,500 picks in an eight hour shift. A suggestion was made in the report to change the AGVs' control algorithms to improve their performance, but the DLA does not have possession of nor the capability to run the original contractor written simulation in order to make the needed changes (3:1,3-11). The lack of a model prevented the DLA from assessing new AGV control algorithms and their effects on the performance goal of the IMC.

Furthermore, many changes have been made to the IMC since the completion of the original contractor study. The storage and retrieval machines have been modified, pickup and delivery stands have been deleted and added in various locations, and the way in which AGVs look for work has been changed. All these changes will impact the performance of the IMC. Thus, the original model has become outdated because of changes in the warehouse design.

### *Problem*

The DLA needs a simulation model to reevaluate the ability of the IMC to meet the needs of its customers. Different AGV control algorithms, the number of AGVs needed, the location and number of pickup and delivery stands, and the modifications to the search and retrieval machines must be examined to evaluate their effect on the performance of the IMC.

### *Objectives*

The purpose of this research was to show the effects of different AGV control algorithms, number of AGVs, location and number of pickup and delivery stands, and the storage and retrieval machines have on the performance of the IMC. The objectives were:

1. Design a model that would simulate the effects of the number of AGVs in the IMC, simulate the storage and retrieval machines to evaluate their performance, and determine if the number of pickup/delivery stands in the IMC is sufficient.
2. Evaluate the results of the simulation with regards to the performance goal of the IMC.
3. Make recommendations, based on findings, to the DLA concerning the effectiveness of proposed or implemented changes in the IMC.
4. Deliver a completed model to the DLA.

## *Scope*

The scope of this thesis was limited to the evaluation of the systems in the IMC that directly relate to the AGV system and the automated storage and retrieval system. Areas of operation in the IMC excluded in this thesis include the rotary racks, conveyor belts, receiving (before module loading), and the packaging and shipping of orders. The program created by this thesis can be used and modified by the DLA to evaluate further changes to the system, such as the effect of adding extra empty modules.

## *Overview of Remaining Chapters*

Chapter II presents a discussion of the literature pertaining to the use of computer simulations, theory on the operation of AGVs and ASR machines, and the validation of simulations. Chapter III describes the flow and control of the AGVs throughout the warehouse complex. Also, the chapter provides an in-depth look at all the operations inside the IMC including receiving, high rise bin and rack storage area, consolidation, shipping and pallet storage area. Chapter IV describes the simulation program and its operation. This chapter includes information on how the AGVs are controlled in the system. Chapter V presents the results of the simulation. Chapter VI discusses the conclusions reached by comparing the results in chapter V to the IMC performance standards. The chapter then offers recommendations based on the conclusions. Appendix A contains the simulation source code written in the SLAM

simulation programming language. Appendix B contains additional fortran code used to support the simulation. Appendix C has the parameter file needed to compile the fortran code and redimension the SLAM program itself. Appendix D contains drawings of the AGV pathways in the IMC. The drawings were used to code the pathways and segments in the simulation source code.

## *II. Literature Review*

### *Introduction*

Experience in the warehousing field shows that buildings are sometimes erected before any thought at all is given to materials handling or storage systems [2:285].

The preceding quote by Williams, reprinted in Ashayeri and Gelders (2:285), states a need to evaluate the performance of materials handling equipment in a warehouse before it is physically installed. Materials handling equipment is used primarily for the moving and storage of materials. Evaluating the performance of the materials handling equipment before installation identifies potential problems and shortfalls which can easily be corrected. The use of computer simulation techniques is one of the most accepted methods of evaluating the performance of materials handling systems.

This chapter focuses on the use of computer simulation techniques to model the performance of materials handling equipment in an automated warehouse, with special emphasis on AGVs and the ASRS.

### *Computer Simulations*

Pritsker draws upon his earlier articles to define computer simulation as a "process of designing a mathematical-logical model of a real system and experimenting with this model on a computer" (15:6). The acceptance of simulation as an analysis tool is growing in the areas of academic and industrial research (5:482). A

simulation model is often used to aid in the selection of the best alternative from a group of proposals (5:482). Simulations allow the analyst to make inferences without physically building, disturbing, or destroying the system being evaluated (15:6). However, the model used in the simulation, must be a reasonable abstraction of a system. Steps to ensure the successful development of a simulation model include: 1) problem formulation, 2) model building, 3) acquisition of the data, 4) translation of the model to appropriate simulation language, 5) verification and validation, 6) experimentation, 7) analysis of the results, and 8) implementation and documentation (15:10-11).

Before a computer simulation is made, the problem must be formulated, a model built, and data collected to be used as parameters for the model. A clear definition of the problem and an explicit statement of the objectives of analysis must be made. The definition of the problem and objectives of analysis provide guidelines for the construction of a model. The model is an abstraction of the problem into mathematical-logical relationships. After the model is built, data must be acquired about the problem. The data is needed to provide parameters of operation for the model. To perform a computer simulation, the model must first be translated into a simulation language (15:10-11).

A simulation language translates the model into a form the computer can understand. The most popular simulation languages on the market today include: SLAM, GPSS,

SIMSCRIPT, and SIMAN (1; 14). Phillips (14:72) classifies simulation languages into different categories such as event orientation and process orientation. He describes event orientation as a simulation that defines the cause and effects of events and measures their impact on the system as they occur. A process orientation would cause the simulation language to view "the world as being composed of sequences of events which occur in a definite pattern" (14:72). Many simulation languages are able to model materials handling systems such as AGVs, cranes, conveyor belts, and assorted robotic equipment (15:590-694). However, only a few simulation languages, such as SLAM, contain special extensions designed to specifically simulate materials handling systems. The extensions contain specialized commands and routines which aid in the programming of the simulation.

#### *Examples of Computer Simulation.*

*Simulation of AGVs.* In performing the analysis of an AGV system, Davis (6) contends that simulation is the best method to use because of its ability to incorporate great detail and its ease of use on a minicomputer. Further, she says that because of the dynamic nature of AGV systems, using a queuing theory based analysis tends to give overly optimistic answers because they do not account for congestion and routing problems. Finally, math programming procedures, many of which are based upon transportation problems, have been presented as solutions to some AGV



design problems. However, Davis points out that the problem size of most industrial applications is prohibitively large, thus requiring very long computer runs. Simulation is used to model three of the most common AGV system characteristics: 1) track layout, 2) the number of AGVs required, and 3) the system control logic (6).

Maxwell and Muckstadt (13) suggest a three stage approach to the design of an AGV system. The first stage consists of gathering relevant data on the vehicle speeds and material flow rates. A note of caution is made to check the data for inconsistencies between departments. The second stage involves the design of the department layout, AGV routes and pickup and drop off points. The third and final stage is broken into two parts: a preliminary pencil and paper simulation; and a computer simulation. The pencil and paper part determines the minimum number of AGVs needed in the system (13). However, this part does not account for time dependent activities such as vehicle blocking and battery charging. The output gives an estimate on the number of vehicles needed for the whole time period (i.e. an eight hour shift). The estimate only provides a lower bound on the minimum numbers of AGVs required, because of the assumption of a perfect system with no delays. But, this lower bound provides a value to compare against the results of the computer simulation portion of the method. The computer simulation part uses a dynamic model to evaluate the aforementioned time dependent activities and the rules

for dispatching vehicles. The output of the computer simulation provides a more exact estimate of the number of AGVs required than the pencil and paper portion. Also, the output contains estimates of the total amount of time the AGVs were blocked and idle. But these estimates do not give any indication of when and where the AGV was idle, blocked, or busy. Also, such statistics as path usage and the number of loads between stands cannot be determined. The pencil and paper part and the computer simulation portion only give a rough point estimate of the number of AGVs required in a system.

Fitzgerald (7:79) and Koff (11) suggest very simple methods to estimate the number of AGVs a system requires. Fitzgerald's formula is:

$$\frac{\text{The sum of all load transit times}}{\text{Traffic factor} * \text{Load factor} * \text{Total vehicle time available}}$$

The traffic factor is a measure of vehicle "competition" for the guidepath (usually .85 to 1.0). The load factor is the percent of time a vehicle is transporting a load. Total vehicle time available is the time period given the vehicles to perform their mission. Load transit time is the time it takes an AGV to take a load from point a to point b. The formula yields an estimate of the number of AGVs needed.

Koff's formula is much simpler:

$$\frac{\text{Vehicle trips/hour required}}{\text{Trips/hour/vehicle}}$$

Vehicle trips/hour required is the average number of times per hour a vehicle is needed. Trips per hour per vehicle is the average number of trips per hour each vehicle can make. Again, the result is an estimate of the number of AGVs the system needs. Again, these results are only optimistic lower bound estimates, because of their assumption of a perfect system.

The DLA has already used a simulation program to evaluate the performance of the IMC. In November of 1985, AutoSimulation Inc. was contracted to write a simulation of the IMC. AutoSimulation chose to write the program using their own AutoMod programming language. The AutoMod language is useful for the collection of such statistics as workstation utilization, AGV utilization, and conveyor utilization. It compiles into GPSS/H, a general purpose simulation language (3:1). Because of the size and complexity of the IMC, AutoSimulation determined it would be impractical to develop a single model of the system. Three smaller subsystems were made: receiving; rack and bin storage, AGV system, and bulk backup storage; and consolidation, packing, and shipping. The final report was completed and published on 31 July 1986.

The AutoSimulations study (3) showed, by means of its simulation, that the DLA's warehouse performance goal would not be met. The study concluded that AGVs were not moving the modules in the most efficient manner possible. A suggestion was made to change the AGVs control algorithms to

improve their performance, but this would require a new study to be written (3:1,3-11).

Davis (6:570-572) outlines control algorithms which have the potential to improve the efficiency of the AGVs in the DLA warehouse. The algorithms are based on such factors as distance, time to arrival, and vehicle utilization rates. The nearest vehicle algorithm selects the closest idle vehicle to the requesting station (6:570-572). This method may decrease the time it takes an AGV to arrive and pickup a load, but distorts the utilization rates of all the vehicles in the AGV fleet. That is, some AGVs will remain constantly busy, while others farther away from the action remain idle longer. A second control method is to select the least utilized vehicle (6:570-572). This method keeps the utilization rates of all vehicles in the fleet approximately equal, but may cause an increase in the time it takes an AGV to arrive at a stand to pickup a load. For example, if the least utilized vehicle is three buildings away from the pickup point, it will take longer to arrive than another AGV in the same building as the pickup point. Finally, AGV use can be altered by using a priority based system in conjunction with one of the two previously mentioned control algorithms. This system accords different priorities to loads. Priorities can be based on first in - first out; last in - first out, and assigned priorities.

Once a vehicle is idle, it must find work. The vehicle can stop where it is and await an assignment, cruise a set

pattern looking for an assignment, or travel to the nearest parking loop and wait for an assignment (15:636-638). The AutoSimulation study used the second method in its simulation (13). However, the IMC will now use the parking loop method to cut down on the number of vehicles cruising idle in the system.

*Simulation of Automated Storage and Retrieval Systems.* An ASRS is composed of storage and retrieval machines used to stock and remove items from aisles of bins and racks. The objective of an ASRS is to maximize the number of "picks" (storage and retrievals) made in a set period of time. Simulation is a better method of optimizing an ASRS than approximating it with a mathematical model (4:288; 8:86). If the "picks" are randomly dispersed, system efficiency decreases because the crane must travel further to get to all the items. Azadivar (4) demonstrates an increase in system efficiency by storing high demand items nearer to the start of the aisle. The less demand for an item, the nearer it is placed to the end of the aisle (4). However, Azadivar does not explain what would happen if the demand for certain parts fluctuated over a short period of time. Han (9) reduces travel time by using a nearest neighbor algorithm. The previously mentioned methods were verified by the use of simulation models (4; 9).

*Advantages and Disadvantages of Computer Simulation.* Although it is agreed upon that simulation study is an

invaluable tool for analyzing various systems (4:288; 6:568; 8; 15), problems can and will arise during the simulation process. Simulation does not necessarily give an optimal or best possible answer to a problem. Simulation gives an answer that is only as good as the design of the model, or to use a common computer term, "garbage in, garbage out."

Hoover and Perry (10) feel that in order to establish the credibility of a simulation, the model must be verified and validated. They define verification as the process of determining whether the logic of the model is correctly implemented, and validation as establishing the model as a credible representation of the real system. To aid the analyst, Hoover and Perry suggest various methods of verifying and validating simulation models (10). Law and McGinnis point out potential pitfalls to avoid in the areas of simulation modeling, simulation software, modeling randomness, and the design and analysis of the simulation experiment (12). Pritsker (15:16-50,723-766) details many of the statistical problems that may affect a simulation. Simulation is a valuable tool, as long as the person using it knows of its limitations.

### *Methodology*

The overall approach of this thesis was to evaluate the performance of the AGV system and the ASR system in the IMC. A review of the literature indicated that a computer simulation model was the best approach (3;4;6;9;13). Simulation languages such as SLAM, TESS, and SIMAN, are

quite capable of handling AGV and ASRS models. These languages were readily available on AFIT computer systems.

### Summary

While there are an assortment methods available to evaluate the performance of materials handling systems, computer simulation is recognized as one of the best methods to use. Computer simulations best feature is the ability to account for time dependent processes. There are many computer simulation languages available that are capable of simulating materials handling systems.

Chapter III presents a detailed description of the flow and control of AGVs throughout the warehouse complex. Also, the chapter provides details on the receiving area; highrise rack and bin storage area, with an emphasis on the automated storage and retrieval system; the consolidation area; and the pallet storage area.

### *III. Warehouse Operations*

#### *Introduction*

The IMC consists of four separate buildings: Receiving, building 110 (fig. 3); the Bin and Rack Storage, building 111 (fig. 4); Consolidation, building 11 (fig. 5); and Pallet Storage, building 10 (fig. 6). The Pallet Storage, Consolidation, and Receiving buildings are all approximately 600 feet in length by 200 feet in width. The Highrise Rack and Bin storage building measures 577 feet in length by 285 feet in width. As shown in fig. 7, the Receiving and Highrise buildings share a common corridor which will be used as an AGV pathway. Covered AGV pathways connect the Highrise building to the Consolidation building and the Consolidation building to the Pallet Storage building.

A Tandem computer is used to control the diverse operations of the IMC. The Tandem computer controls the conveyor system, rotary racks, the location of items in the highrise and pallet storage area, and processes requests for AGVs. The flows of modules and pallets in the IMC are found in figures 8 and 9 respectively. The actual handling of the AGVs is done by a secondary computer called the AGV computer.

#### *Performance Goal of the IMC*

The performance goal of the IMC is to process 15,500 pick items per day in the consolidation area. The pick items are items taken from the storage racks in the



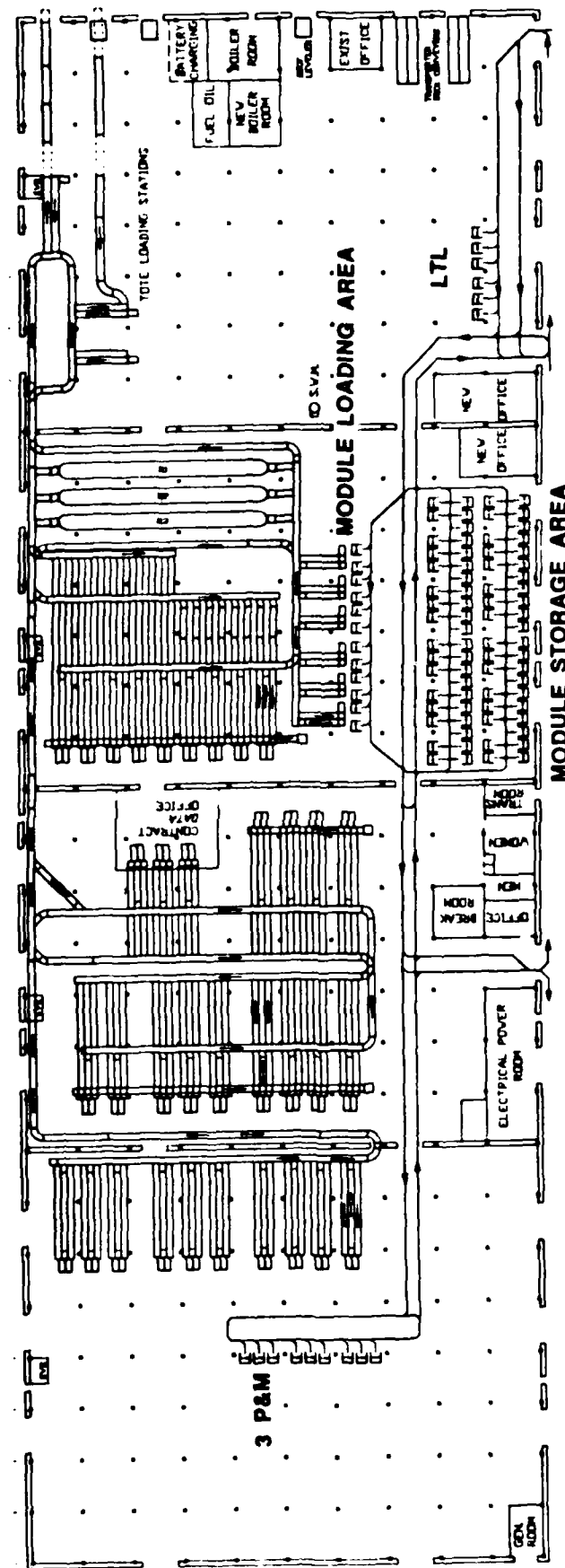
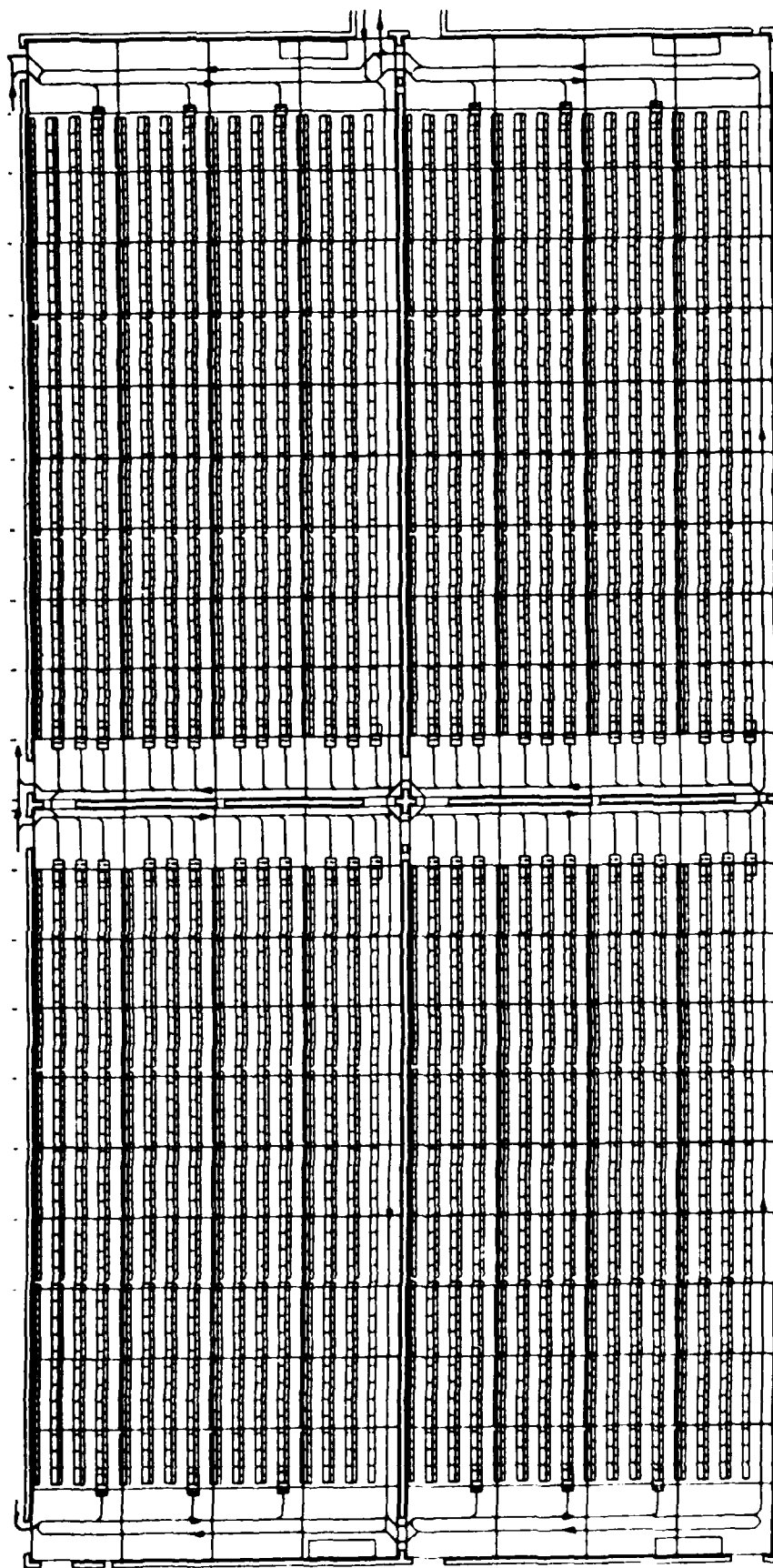


Figure 3. Layout of the Receiving Building



**Figure 4. Layout of the Highrise Rack and Bin Storage Building**

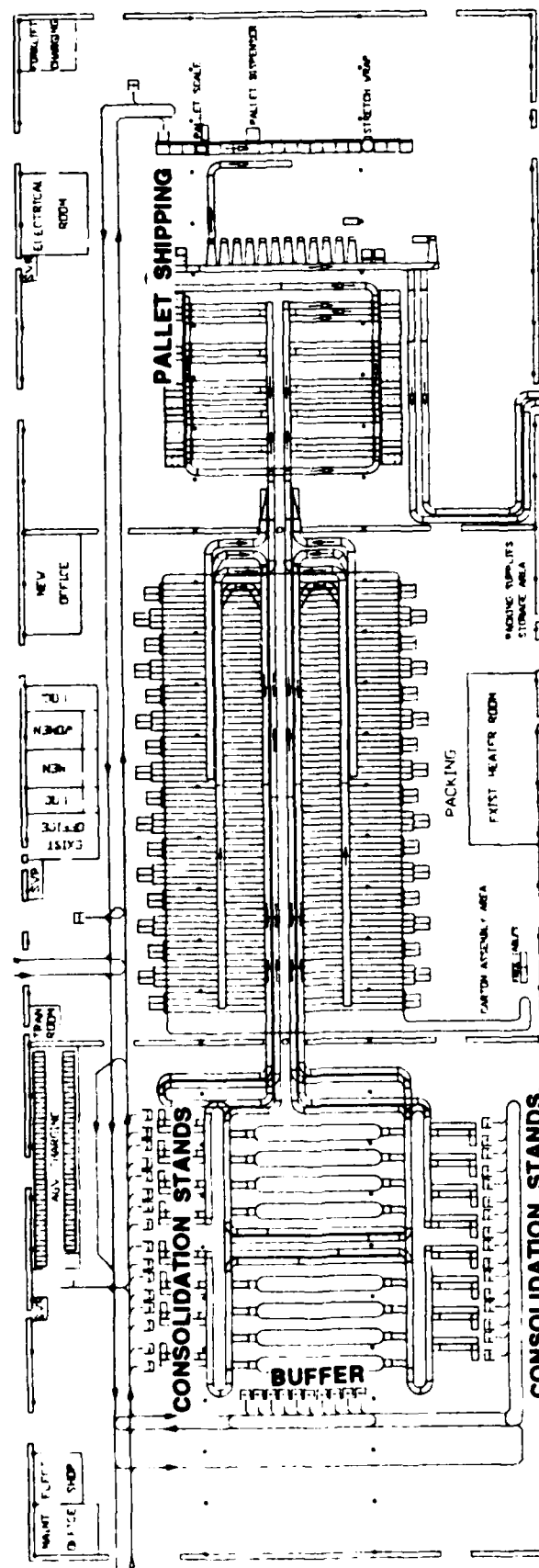


Figure 5. Layout of the Consolidation Building

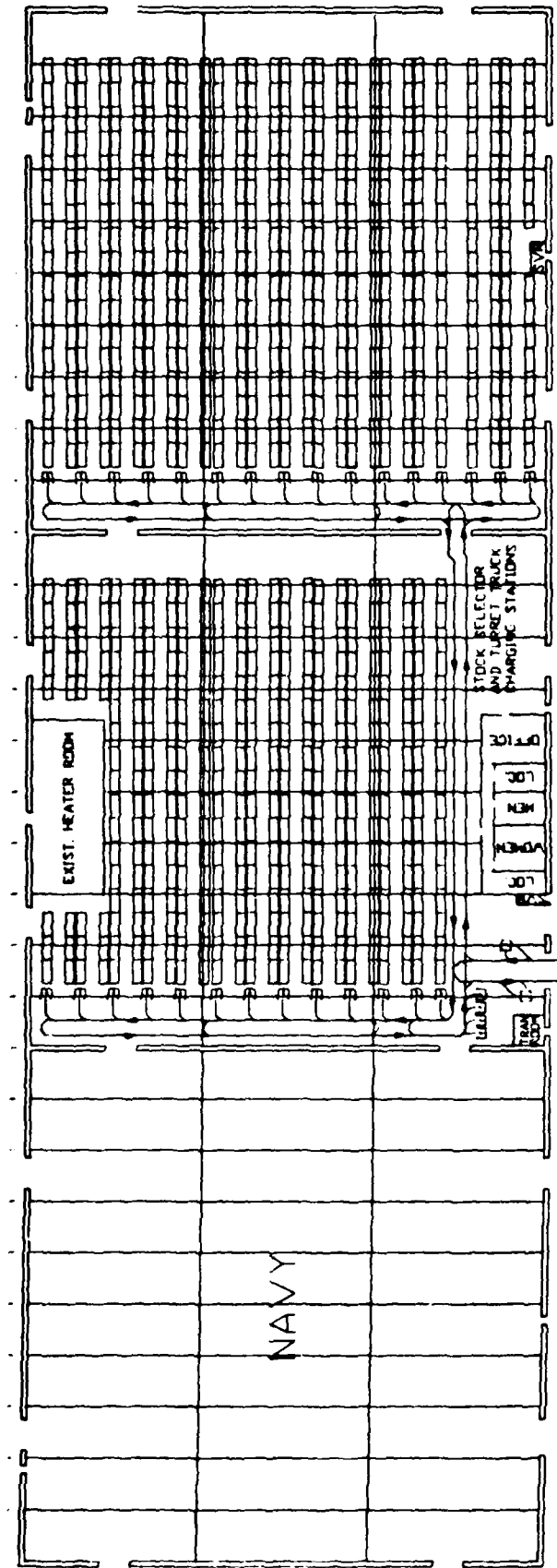
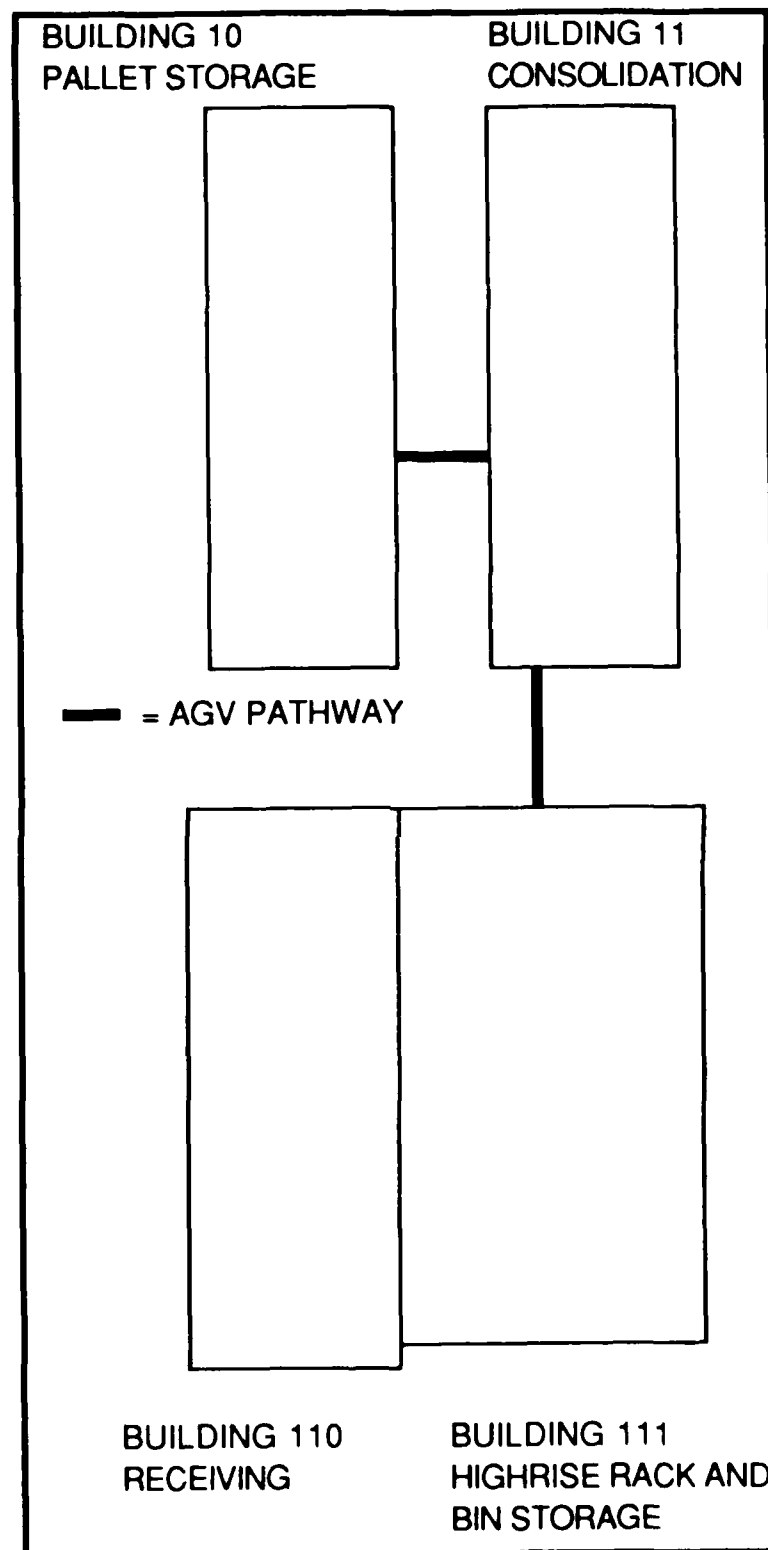
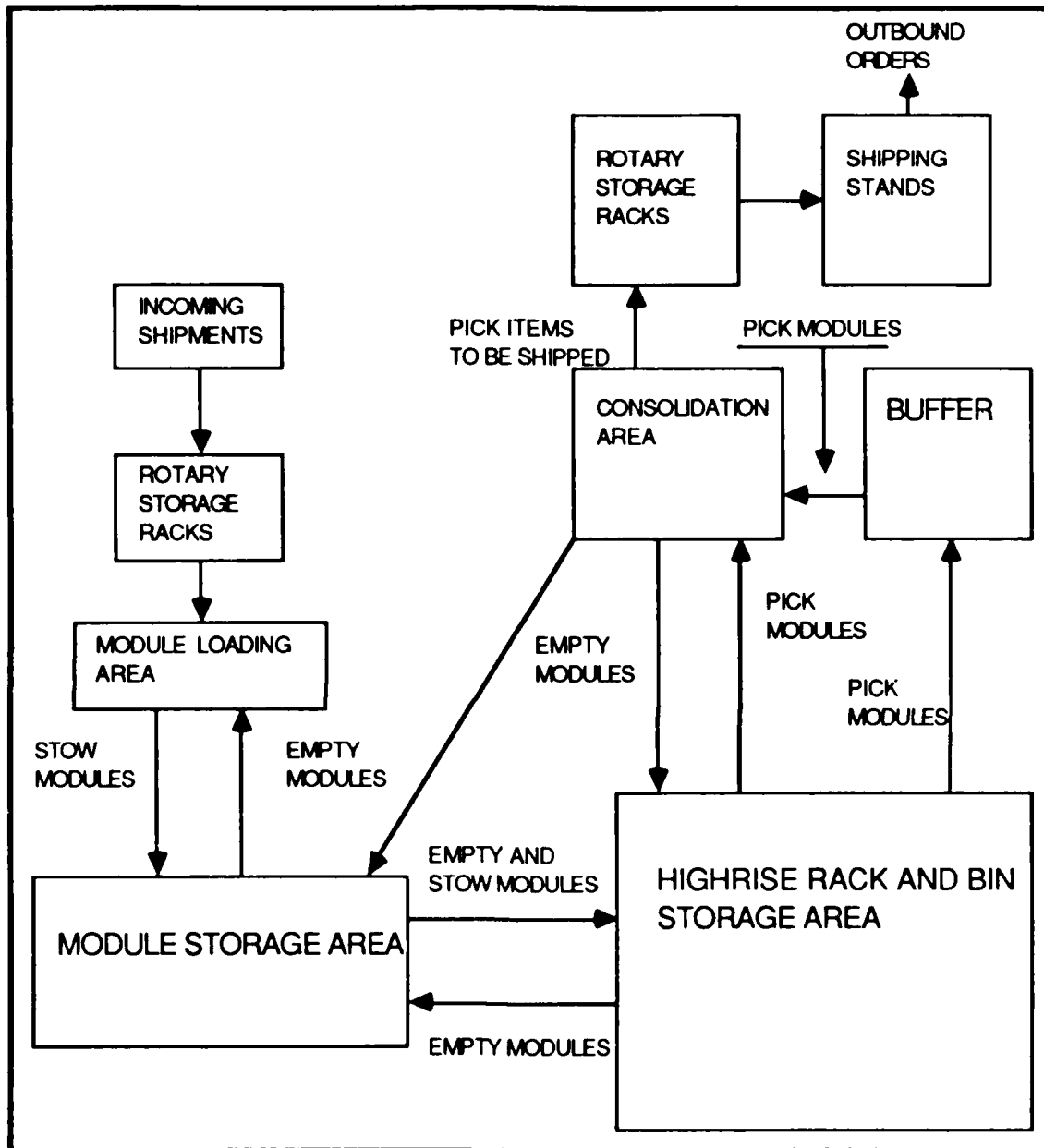


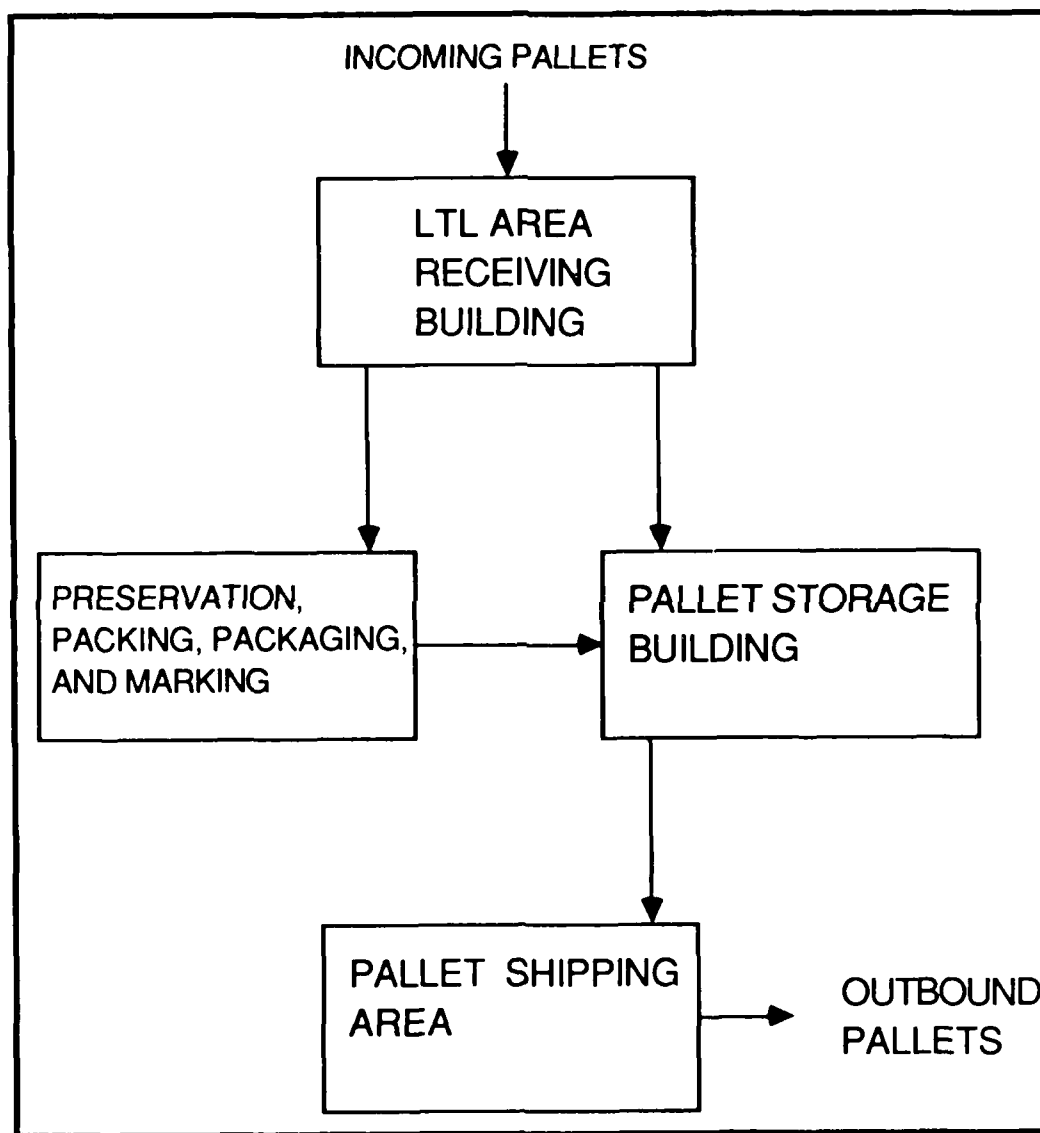
Figure 6. Layout of the Pallet Storage Building



**Figure 7. IMC Layout**



**Figure 8. Flow of Modules in the IMC**



**Figure 9. Flow of Pallets in the IMC**

highrise. They are then transported to the consolidation area in modules by AGVs. The pick items are then consolidated into orders and prepared for shipping.

#### *Module and Pallet Description*

The AGVs in the IMC are used to transport empty or full modules or pallets to various locations throughout the four building complex. A module is a steel and wire frame box measuring 40" wide by 48" deep by 57" high and can hold 20 to 30 items on each of its two shelves. Modules are described as empty, stow, pick, or full modules. Empty modules are just that, empty. Stow module are modules that have been filled with items at the module loading area to be stowed in the high rise bin and rack storage area. A pick module is a module that has been filled with items from an aisle in the high rise bin and rack storage area to be taken to the consolidation area to be processed for shipping. A full module is just another name for a module containing picks or stows.

A standard pallet is used for larger, bulkier items and measures 40" wide by 48" deep by 8" high. Pallets are generated in the LTL area of the Receiving building. The majority of pallets are taken to the pallet storage area. A small percentage of the pallets are taken to the 3P&M area for additional processing and then taken to the pallet storage.



## AGVs

*Description.* The AGVs will be built by Warehouse Handling Systems of New York. The AGV fleet will consist of 25 identical vehicles. Options exist to purchase up to 15 additional vehicles in lots of five, if needed to meet the processing goal. Each vehicle is 96" in length (including two 18" bumpers), 43" wide, and 26" high. The top speed is 4 feet per second with an acceleration/ deceleration rate of 3 feet per second squared. So, for example, an AGV accelerates to its top speed in 1.3 seconds.

The propulsion source is an electric motor powered by rechargeable lead acid batteries. During a normal eight hour shift, approximately 75% of the AGV fleet will require recharging. The discharged battery will be replaced with a freshly charged battery in less than five minutes, rather than taking the vehicle off line for a longer period of time to recharge its own batteries. An AGV with a low battery signals its condition to the Tandem computer, which in turn directs the AGV computer to send that AGV to the battery replacement station in the consolidation area as soon as the AGV is free.

*Pickup and Delivery.* To pick up a module or pallet at a pickup/delivery (pd) stand (dimensions 40"w by 48"d by 30"h), the AGV drives under the stand, stops, and starts raising its top until the pallet or module is lifted off the stand. Next, the AGV backs out from under the stand, lowers its top, and drives off. To deliver a module or pallet, the

procedure is reversed. The unload/load process takes approximately 30 seconds after the AGV comes to a halt.

*Guidewire Paths.* The AGV receives guidance from wires buried in the floor. The wires transmit a low frequency signal which the AGV follows from point to point. An AGV cannot move if it leaves the wire path. Approximately 10,760 feet plus or minus 300 feet of wire pathway will be installed in the IMC. The pathways (fig. 10) consist of

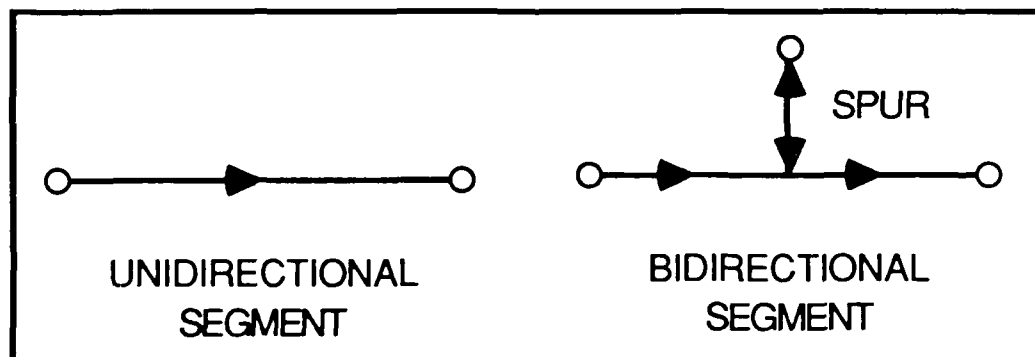


Figure 10. Pathway Example

unidirectional segments and spurs. Unidirectional segments permit AGV travel in one direction only. Spurs are bidirectional paths that allow the AGV to drive in, deliver or pick up a module, then back out.

*AGV Control and Anti-collision Measures.* An AGV receives its orders from a centrally located computer. First, a worker sends an order for the AGV to bring or take away a module to the Tandem computer. The Tandem is the computer used to control the whole complex. The AGV order is passed from the Tandem to the AGV computer, solely

designed to control the AGVs. The AGV computer keeps track of AGV orders and the status of all AGVs in the system. An AGV can be in one of the following states: idle (no order); traveling to a parking loop; idle, parked at a parking loop; traveling empty to pick up a module to take somewhere; and, traveling full with a module.

The AGV computer selects the idle AGV nearest the location where the order was sent and gives it the order. The AGV then travels to its destination by the shortest route method. Each AGV has an internal memory containing a map of all the pathways in the complex to use for planning its route. However, the AGV is not able to avoid congested areas, which may appear along its path since it has no knowledge of the location of the other AGVs in the fleet. In the event of a traffic, the AGVs involved would have to manually separated by the workers in the IMC.

To prevent collision between AGVs and to allow the AGV computer to communicate and track the AGVs, a series of control points are used. Control points are placed around every intersection and approximately every 20 feet on long segments. When a control point senses an AGV passing, it becomes activated. The control point will stay activated until the same AGV passes another control point. If another AGV attempts to pass an activated control point, it will be stopped. For example, in fig. 11, a passing AGV would activate control point A. When the AGV passes control point

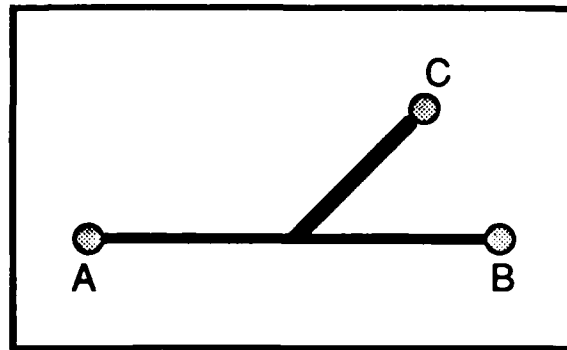


Figure 11. Control Point Example #1

B or C, control point A would be deactivated. However, if the AGV stopped before reaching control point B or C, control point A would stop a second AGV from colliding with the first AGV. In figure 12, an AGV traveling from A to B

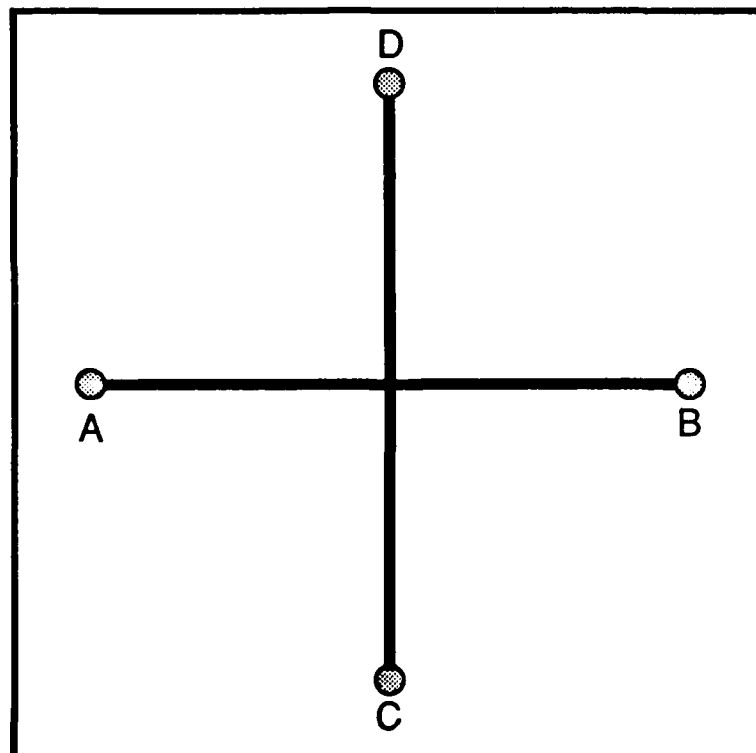


Figure 12. Control Point Example #2

would activate control point A and C to prevent itself from being hit from the rear or side. Upon reaching control point B, control points A and C would be deactivated.

The AGV computer keeps track of the AGV fleet by using the control points. Each time an AGV passes a control point, it sends a message to the computer through the control point. The computer knows where all the AGVs are by means of the last control point each AGV passed. To give an AGV an order, the computer sends a message to the control point the AGV is at or about to pass. The computer cannot communicate with an AGV travelling between control points.

*Parking Loops.* When an AGV is finished with its job and no jobs are pending, the AGV will drive to the nearest parking loop and stop until selected for an assignment. A parking loop is a pathway where AGVs can go and stop without interfering with the other AGV traffic. A system without parking loops causes the AGV to either stop at a control point, thereby blocking traffic, or cruise needlessly looking for work, also causing blockages and wasting battery power. A previous design of the IMC used in the Autosimulation study (3), used the former method of AGVs looking for work. There are four parking loops now planned for the IMC. Two in the receiving area and two in the consolidation area.

#### *Receiving*

Incoming shipments are directed to the Receiving building (fig. 13). There, the shipment will be loaded into

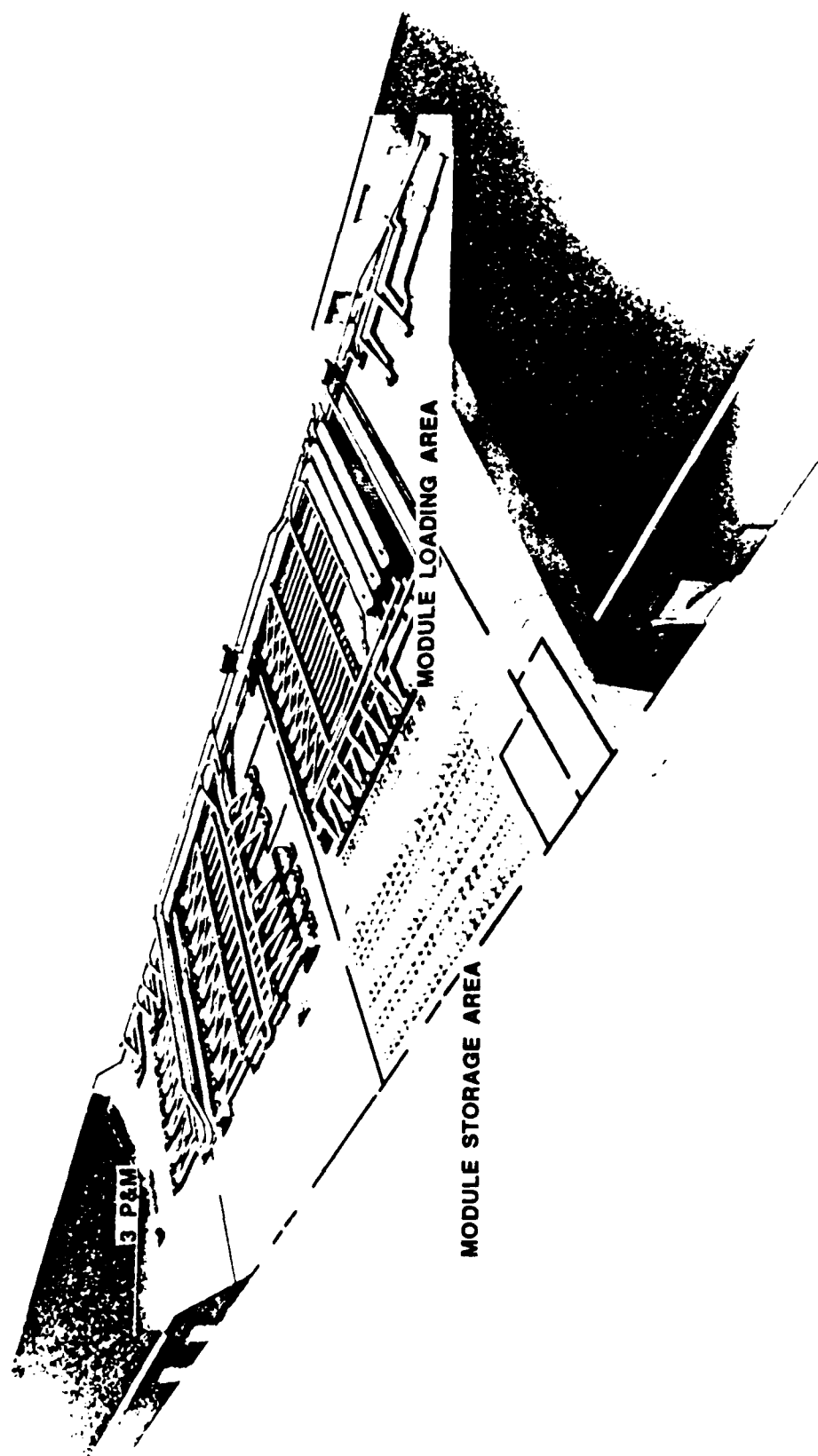


Figure 13. Receiving Area

totes (plastic bins) by the workers and placed onto the conveyor network. The conveyors take the totes to various stations where workers unpack, inspect, and if necessary, repack the items. From there, the items will be placed in a rotary storage rack until enough items destined for one row in the highrise are accumulated to fill a module to be sent to that particular row. The Tandem computer determines when there are enough items to make a load. When a load is ready, the items are sent to a lane in the module loading area.

*Module Loading Area.* There are six workstations in the module loading area, an increase of two since the Autosimulation study was performed. Each station has one worker and is equipped to handle two separate loads and two modules at a time. A load is a group of items all destined to be stowed in the same row in the highrise. A load consists of 20 to 30 items. Although a module can handle more items, the bottom shelf is intentionally left empty to allow the storage/retrieval machine operator to simultaneously pick items into and stow items from the module. When enough items have been accumulated in the rotary rack, the Tandem computer checks the workstations to see which has an empty module ready for loading. After an empty module is located, the load is transported to the workstation by a series of conveyor belts. At the workstation, the worker loads the module and sends an order to the Tandem for an AGV to take the stow module to the

module storage area. When an AGV arrives to pick up the stow module, the worker sends another to the Tandem for an AGV to bring the workstation an empty module from the module storage area. A stow module leaves a workstation about every 6.25 minutes. Approximately 80 stow module are created in one 8 hour shift.

*Module Storage Area.* The module storage is used to store stow and empty modules until they are needed. There are not any workers present or required in the module storage area. In the Autosimulation study (3), this area consisted of 121 pickup/delivery stands. Now, only 74 stands are planned for in the IMC. Stow modules are sent here from the receiving workstations until they are called for by a storage/retrieval machine operator in the high rise. Empty modules in the storage area are supplied to the receiving workstations and the rows in the high rise for filling. Empty modules are created in the consolidation area. After a module is emptied, the system attempts to send it to a high rise row. If that is not possible, the module is sent to the module storage area.

#### *Automated Storage and Retrieval System*

*Description.* The high rise bin and rack storage facility (fig. 14) is made up of four identical sections of 15 aisles each. An aisle is 232 feet long and is 40 feet high. It is divided into 40 one-foot high tiers by 76 three-foot wide bays, for a total of 3,040 storage locations per aisle. At one end of each aisle are two module stands



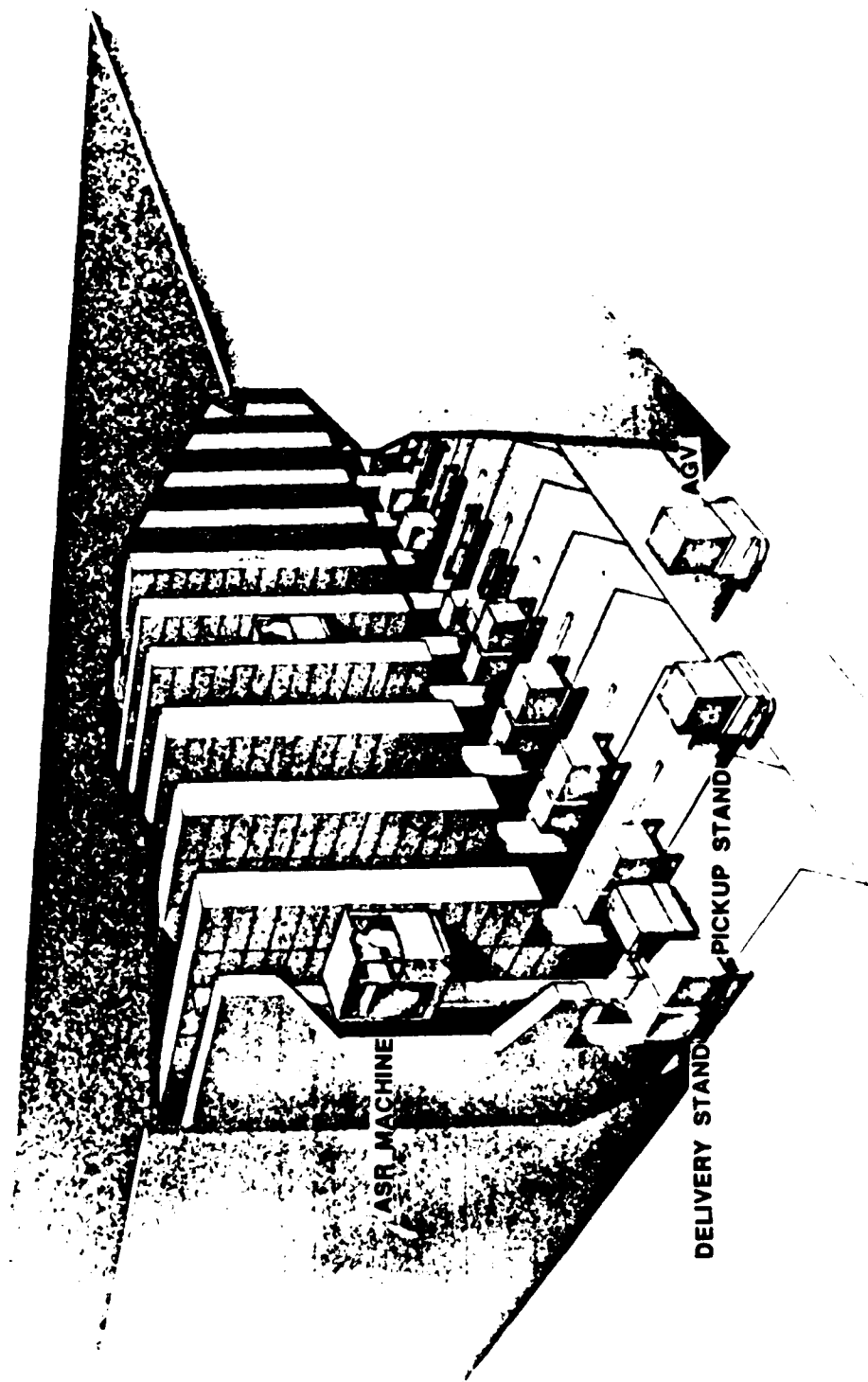


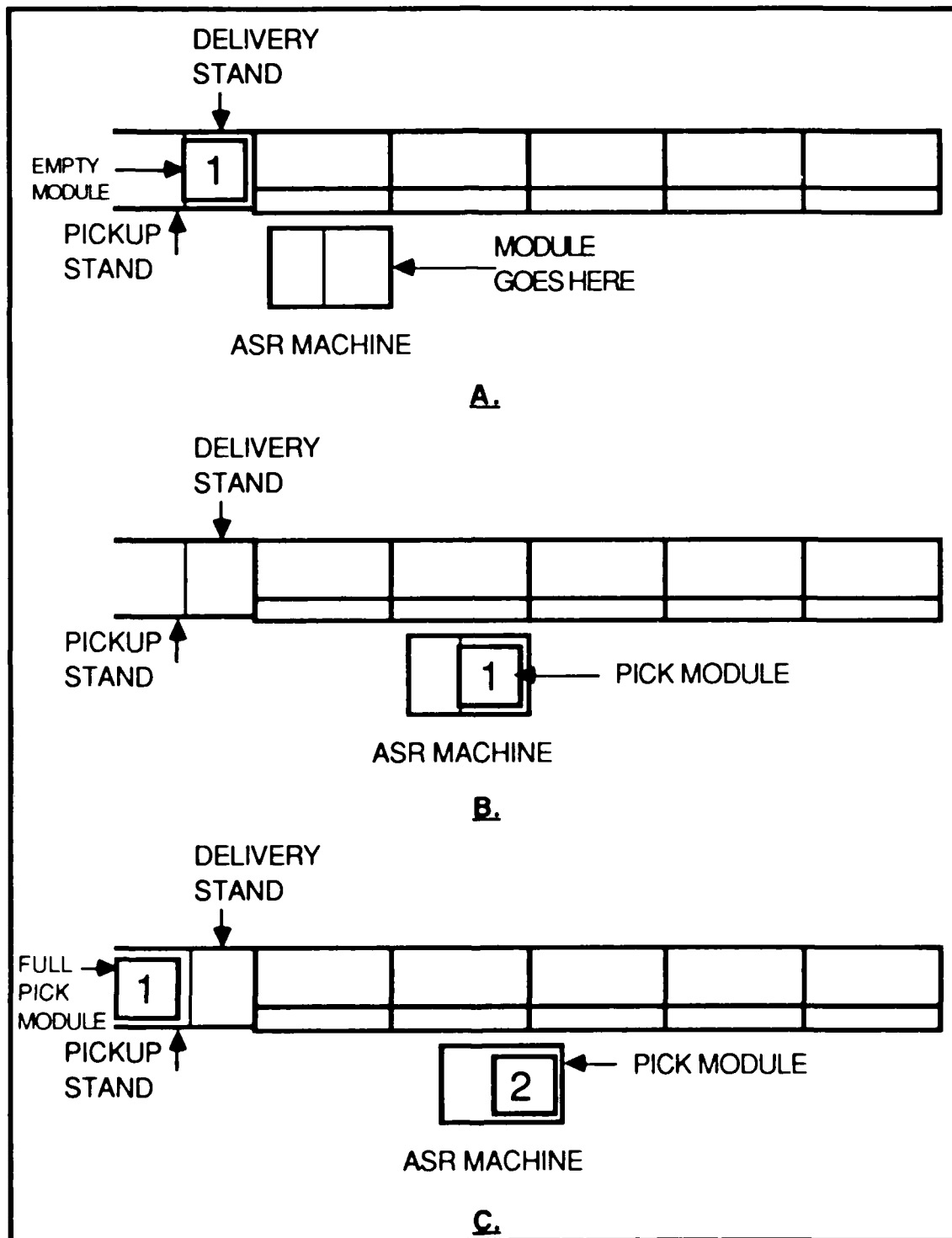
Figure 14. Highrise Rack and Bin Storage Area

in series, the inner one exclusively for deliveries and the outer one for pickups. The delivery stand supplies modules to the man on board storage/retrieval (S/R) machine. The S/R machine picks up a module and is guided to a particular storage location where a pick or stow is to be made. The crane operator receives directions on which location to go to from the Tandem computer. Each of the 60 aisles have an S/R machine. The speed of an S/R machine is 440 feet per minute horizontally and 60 feet per minute in the vertical direction.

*Picking Priorities.* At the start of each day, the operator of each S/R machine must make a certain number of picks. The picks are classified into three issue priority group (IPG) categories: IPG I, IPG II, and IPG III. Each aisle will make approximately 258 picks per day, for a total of 15,500 picks. The goal of the IMC is to process all 15,500 picks in the consolidation area in one day. Items classified as IPG I must be picked first. There are two ways to pick IPG I and IPG II items. The first method is to fill a module with all the IPG I items and send it immediately to the consolidation area. Then, the next module is used to pick IPG II items. Additional modules may be needed to pick all of the IPG II items. The second method is a combined IPG I/II pick run. If there is any room left in the module after all the IPG I items have been picked, the operator will fill it to capacity with IPG II items.

IPG III items are picked only after all of the IPG I and IPG II items have been picked. At that time, the S/R operator checks to see if a stow module destined for his row is in the module storage area. If there is, the operator tells the Tandem to bring him that module. When the module arrives, a combined pick (IPG III items) and stow (items in the stow module) run is made. Since the module is half full, only half the normal number of picks can be made. Because the locations of the pick items and the stow items are randomly dispersed throughout the aisle, it would be very inefficient timewise to make all the stows first and then pick into both shelves of the module. If there are not any stow modules available for that row and there are still IPG III picks remaining, a call is made to the Tandem for an empty module.

*Operation.* Before the start of the day shift, an empty module is placed on the delivery stands of every aisle (fig. 15a). The S/R machine picks up the module and the operator notifies the Tandem to send another empty (fig. 15b). The S/R machine cannot set the first module down in the pickup stand until the second module is delivered, because a module in the pickup stand blocks the AGV delivering the second module. When the first module is filled and the second module has been delivered, the S/R machine puts the first module in the pickup stand and picks up the second module to begin filling it with picks (fig. 15c). When the first module is ready to be taken away, the operator signals the



**Figure 15. ASR Machine Example**

Tandem that an AGV is needed. Before the Tandem can tell the AGV computer to send an AGV, it must find an available workstation in the consolidation area to take the pick module to. Until a workstation is found, the first module must wait on the delivery stand. When a workstation is found, the Tandem tells the AGV computer to send an AGV to pickup the module. After the AGV arrives and takes the first module away, the operator makes a call to the Tandem for an AGV to bring an empty module. This process will be repeated over and over until all the IPG I, II, III picks are made and stow modules are emptied.

#### *Consolidation*

The consolidation area (fig. 16) processes the pick modules brought to it from the high rise storage area. It is made up of 32 workstations, each with one worker, and 10 buffer stands. The workers at the workstations unload pick modules and the buffer stands hold the extra pick modules if all the workstations are busy. After the picks are unloaded from a module they are taken by conveyor belt to the packing area.

The packing area has 68 workstations, each with one worker, to pack each item for shipping. Next, each item is sent by conveyor belt to the rotary storage racks. The rotary storage racks are used to consolidate orders. When all the items in an order have been picked and placed in the rotary racks, the Tandem releases the order to the shipping area. The majority (78%) of the orders are for less than 10

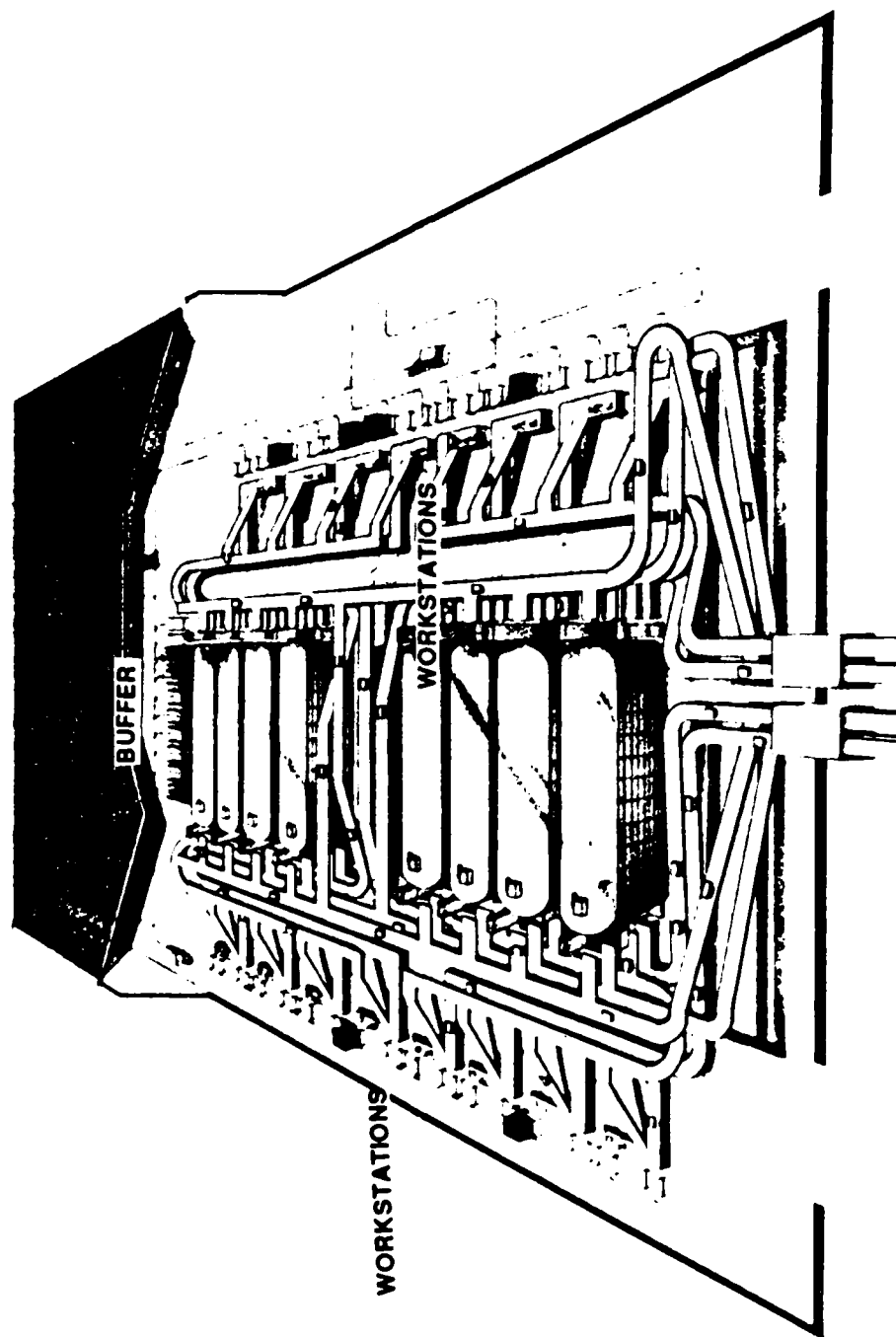


Figure 16. Consolidation Area

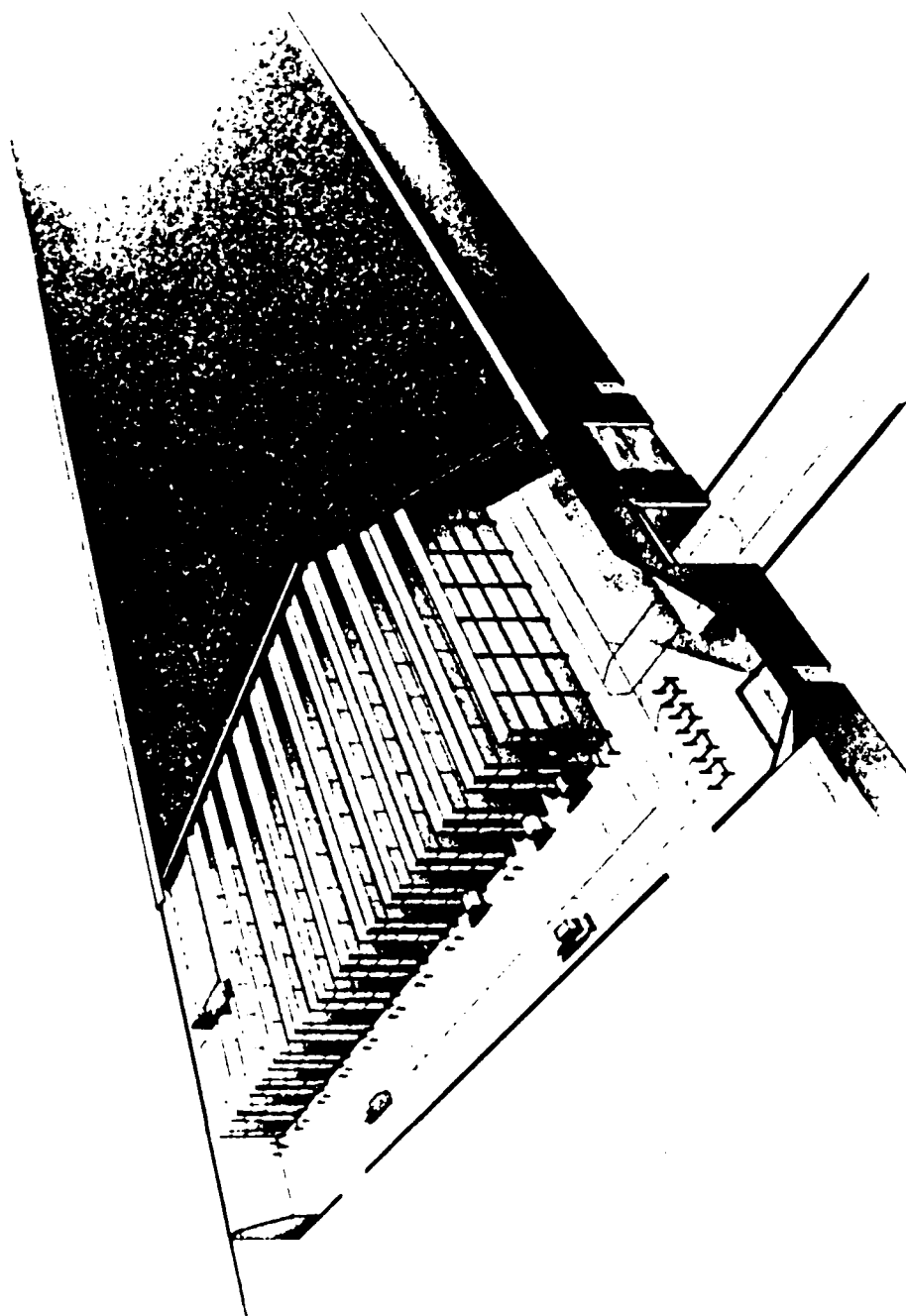
different types of items, 19% of the orders are for more than 10 but less than 100 different items (3). Workers at each of the 34 shipping workstations then process the items for shipping. The last step in the process is to take the order to the shipping docks.

#### *Pallet Movement*

Pallets are used for large loads of identical items such as cartons of paper, or items too large to fit in the highrise storage area. Pallets of items are treated differently than modules in the IMC. Pallets are unloaded from trucks in the North end of the receiving building by forklifts. The forklifts place them on the LTL (less than a truck load) pickup delivery stands that are reserved for pallets only. When a pallet is placed on the stand, a call is made to the Tandem for an AGV to take the pallet to the pallet storage area or the preservation, packing, packaging and marking (3P&M) stands.

The pallet storage area (fig. 17) has 26 aisles of storage with 17 bays in each aisle with four tiers in each bay (3). Incoming pallets are dropped off on a stand at the end of each aisle. From there, one of six turret trucks in the building will pickup the pallet and move it to its proper location. Again, the Tandem gives the location of the bay to the turret driver.

The 3P&M stands are used to inspect, repackage, repack or remark items on the pallets. Further, a load on the pallet may be broken down into smaller lots to be stored in



**Figure 17. Pallet Storage Area**



the highrise. After the work is completed, a call is made to the Tandem for an AGV to take the pallet to the pallet storage area.

If a pallet needs to be shipped out, the Tandem notifies the driver of a turret truck to move the pallet to the pickup/stand. After the pallet has been found and loaded on the stand, a call is made for an AGV. The pallet is taken by the AGV to a special pallet processing stand in the consolidation area. Here, the pallet is readied for shipping.

#### *Summary*

The movement and flow of items in the IMC are controlled by the Tandem computer. From the moment an item is introduced into the system, its location is tracked by the system. Conveyors are used to transport individual items into receiving and from the consolidation workstands to the shipping areas. AGVs are used to transport module loads of items between receiving, the highrise, and the consolidation area. AGVs are also used to transport pallet loads about the IMC.

Chapter IV will present the description of the simulation program written to simulate the functions of the IMC described in this chapter. The emphasis of the program is on the AGVs, ASR machines and the flow of modules and pallets in the IMC.

#### *IV. Program Description*

##### *Introduction*

This chapter shows how the processes described in Chapter III were translated into a computer simulation. First the choice of methodology will be discussed. Then program used to simulate the IMC processes will be described.

##### *Methodology*

At the request of the sponsoring agency, DLA/DORO; a computer simulation of the IMC was written using the Simulation Language for Alternative Modeling II (SLAM II) with the Materials Handling Extension (MHEX). The literature review (Chapter II) confirmed that a computer simulation was indeed the best approach to the problem. SLAM II with MHEX is a powerful computer simulation language which offers more in-depth and precise information than the pencil and paper methods of Muckstadt and Maxwell (13), Fitzgerald (7:79) and Koff (11).

SLAM II with MHEX is capable of simulating the continuous movement and monitoring the work status and location of each individual AGV in the system. These features allow the programmer to find out such things as where and when traffic congestion occurs, segment usage, where the AGVs travel to the most, and the productivity (efficiency) of the AGV fleet. Since the simulation is a continuous process, virtually any time dependent statistic

such as the aforementioned features, can be measured. Pencil and paper methods cannot accurately model time dependent data; SLAM II can.

The data collected each time the simulation is run only reflects the conditions of that run. Multiple runs must be made and the mean and variance of the data evaluated to ensure the accuracy of the whole simulation. Pencil and paper methods are only point estimates without any variance estimates, hence the accuracy of the answer cannot be easily verified.

A simulation allows the programmer to ask many "what if" questions. For example, if the programmer desires to find out what happens when an AGV breaks down and blocks traffic at a key intersection during the fifth hour, he can by using a simulation. But with the pencil and paper method one could not because that level of detail is not possible.

A computer simulation is not the answer to all questions. It can only answer specific questions about the system it was designed to represent, not any other system. A pencil and paper because of its simplicity can be applied to many different systems with little or no modifications. Which method to use depends on the level of detail desired by the user. If a "quick and dirty" answer is needed, then a pencil and paper method is best. However, if specific details are needed, including time dependent processes, then a computer simulation is the recommended (6; 15).

### *Description*

The simulation of the IMC consists of a SLAM II data file and a fortran program containing additional subroutines. The fortran subroutines are used to initialize the simulation (INTLC), define and execute complex events (EVENT), and output data (OTPUT).

The INTLC subroutine initializes the simulation. It resets counters and sets up the warehouse to begin the simulation. During the simulation, a need sometimes arises for an action to be taken that is too complex to be described in SLAM. The subroutine EVENT allows these processes to be defined in fortran code. When one of the actions described in the subroutine needs to be taken, it is called by the SLAM program and executed. The OPUT subroutine outputs data and information from the simulation.

A listing of the SLAM code and fortran code can be found in the appendices. Should the reader desire a more detailed description of the programs, it is suggested that appendices be consulted.

### *Defining the AGVs*

In order for SLAM II to simulate an AGV, it needs to know certain performance parameters and measurements of the AGV. An AGV is defined using the VFLEET command. The statement contains information about the AGVs. The name of the vehicle fleet in the simulation was AGV. The number of vehicles in the system was varied from 20 to 35 in increments of five. Each vehicle has a top speed of

four feet per second (2.72 miles per hour), regardless of whether it is loaded or not. An AGV accelerates and decelerates at the rate of 3 feet per second squared. When SLAM moves a vehicle, it accelerates it to top speed and decelerates the vehicle when it comes to a stop. The acceleration and deceleration feature adds to the realism of the simulation such as when an AGV must stop suddenly to avoid a collision. The length of an AGV is 5.0 feet, excluding bumpers (fig. 18). The closest distance two AGVs

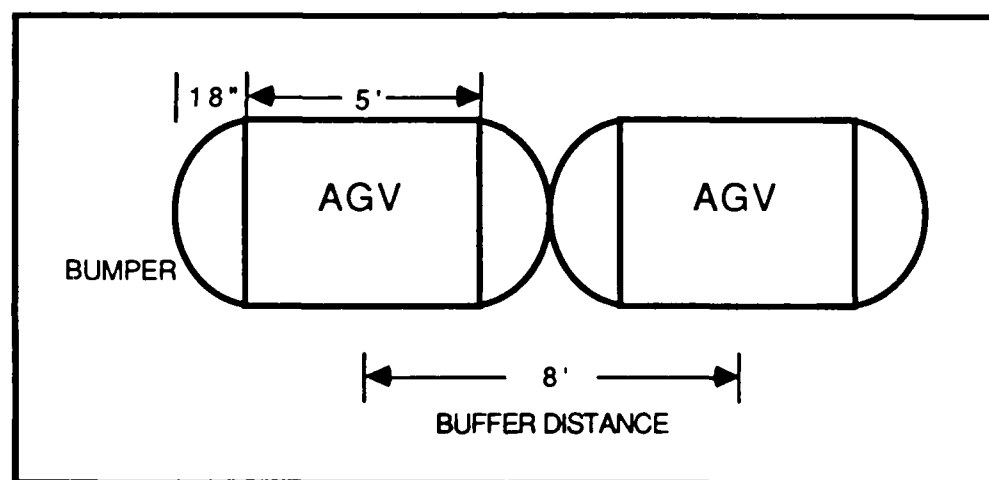


Figure 18. AGV and Buffer Distance

can come together is eight feet. This distance is called the buffer distance and is measured from the center of one vehicle to the center of the other. This parameter is used to prevent collision between AGVs. If SLAM detects a moving AGV is about to collide with a parked AGV, it will decelerate and stop the vehicle at the buffer distance from the parked AGV. The checkzone parameter defines the closest distance, 2.5 feet, around an intersection to which an AGV

may come if the intersection is occupied by another AGV. An intersection is occupied by the first AGV to come within 2.5 feet of the center of the intersection. In fig. 19, vehicle

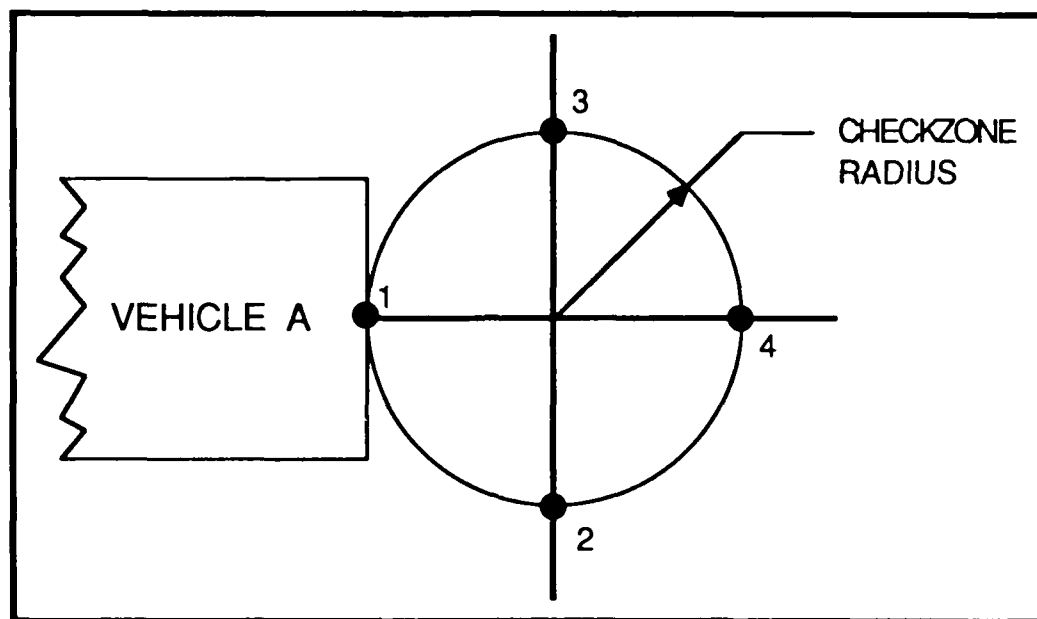


Figure 19. Checkzone Example

A has entered the intersection by coming within 2.5 feet of the center of the intersection (point 1). Until vehicle A has passed either point 3 or point 4, each 2.5 feet from the center of the intersection, a second vehicle B, would stop and wait at point 1 or point 2. The checkzone parameter helps to prevent collision at intersections. However, if an AGV breaks down in the middle of an intersection, it will continue to block traffic until it is removed. In the IMC, when a vehicle becomes idle, it is directed to go to a parking loop. When a vehicle in the simulation is idle, it

searches the list of parking loop locations, finds the closest and proceeds to travel there. If an order comes in before it reaches the loop, the vehicle is diverted. When an idle vehicle reaches the parking loop, it stops and awaits the next order. The last field specifies the number of AGVs to be placed at various locations in the complex at the start of the simulation.

#### *Track Layout*

In the real IMC, the AGVs follow a buried guide wire to their destinations. These guidewires form a complete network of segments and nodes (intersections). The materials handling extensions duplicate this network by using the VCPOINT and VSGMENT commands. VCPOINT defines the intersections (control points) in the network. Control points are intersections and endpoints (fig. 20,21). An AGV

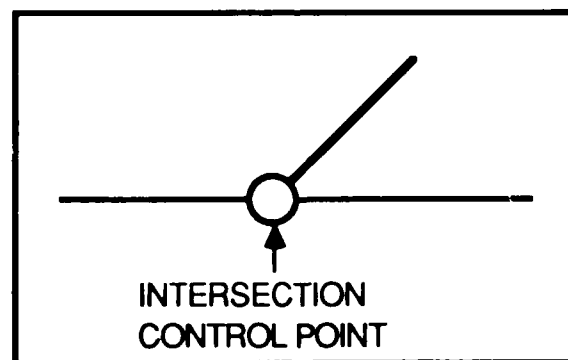


Figure 20. Intersection Control Point

can be directed to travel to any control point in the network. VSGMENT defines the pathway between the control

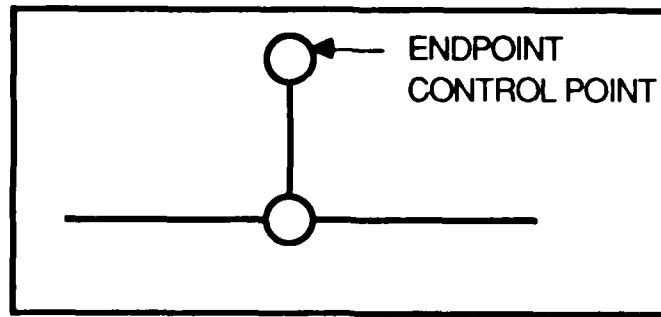


Figure 21. Endpoint Control Point

points. A segment has a starting control point, ending control point, distance, direction and capacity. The segment can be unidirectional, in which case the AGV can only travel from the starting control point to the ending control point, or bidirectional. For this simulation, the only bidirectional segments were the spurs the pickup and delivery stands are located on. Bidirectional spurs allow the AGV to drive into the stand and back out.

The IMC simulation required 453 control points; 242 were pickup and delivery stands, 4 were parking loop destinations, and the rest were intersections of segments. The same simulation required the defining of 553 segments; 242 spurs for pickup and delivery stands, 4 parking loop segments and 307 miscellaneous segments. The data used to define the control points and segments was collected by the author using a pair of dividers, an architects scale and 1/50" to 1' scale drawings of the AGV pathways in the IMC.



The minimum length of a segment is defined by SLAM as the length of the vehicle plus two times the checkzone radius. For this simulation, the minimum segment length was ten feet. However, many of the actual segments in the IMC are less than 10 feet. In order to accurately simulate AGV movement a compromise was made. Figure 22 shows a portion

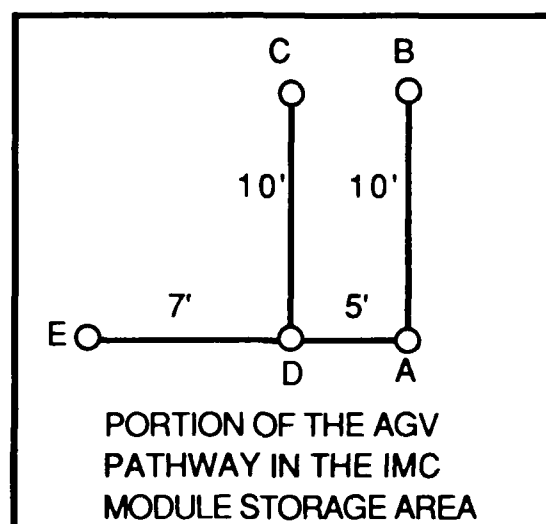


Figure 22. IMC Segment

of the module storage area as it looks in the IMC plans. The control points are labeled A through F and the distances between each are marked in feet. SLAM cannot model this section because segment AD is five feet long and segment DE is seven feet long. In the real IMC, an AGV entering or exiting segment would block an AGV from entering or exiting segment CD. This is because the control points are activated to prevent collision. An AGV entering or exiting segment CD would do the same to an AGV exiting or entering

segment AB. The simulation compromise is presented in fig 23. Control points C and B are now connected to A and

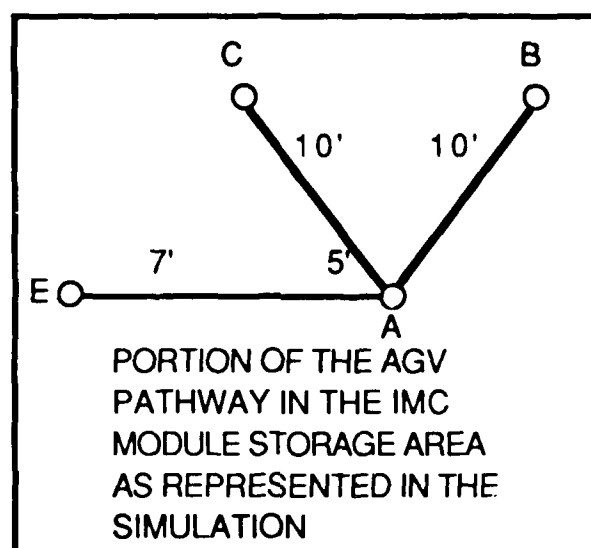


Figure 23. Simulation Segment

control point D is removed so segment AE is now 12 feet long. An AGV entering segment AB will still block an AGV from traveling to or from point C and vice versa. The only effect this compromise has on the simulation is very minor. In the IMC an AGV delivering a module would travel over segments AD and DC for a total of 15 feet. In the simulation, the same vehicle would travel five feet less. This causes the total time that the AGV was considered to be working to decrease by 1.25 seconds, and the idle time of the vehicle to increase by a like amount. But, because the vehicle is working for tens of thousands of seconds, this small amount will not affect the statistics that show the

percentage of time a vehicle is idle or working. Other segments such as U-Turns were lengthened to ten feet under the same argument that the extra time it would take an AGV to traverse the extra distance would be insignificant.

#### *Pickup and Delivery Stands*

The location of a pickup and delivery stand is represented by a control point at the end of a spur. Resources are used to represent the status of pickup and delivery stands in the receiving area, module storage area, and consolidation area. If a module occupies a stand, the resource belonging to that stand is placed in use. A resource that is in use prevents another module from being brought to that stand. This prevents two or more modules from being placed on a stand made for one module.

#### *AGV Movement*

AGVs move from point A to point B by the shortest path possible, regardless of traffic congestion or blockage. At the beginning of the first simulation run, SLAM creates a matrix containing the shortest path information from and to every control point in the simulation. With 453 control points, at least 200,000 shortest pathways must be calculated. Needless to say, this part was the most time consuming portion of the simulation.

Modules are represented by SLAM entities, each defined by nine attributes (table 1). Attributes may be thought of as labels attached to the module. Each label contains a

Attribute #	Description
1	Present control point location of module
2	Module storage area destination
3	Other destination control point
4	Number of stows inside the module
5	Number of picks inside the module
6	Flag or type of order
7	ID number of location resource (the module occupies this stand)
8	ID number of destination resource (the module will occupy this stand)
9	Time the module was picked up

Table 1. Module Attributes

pertinent piece of information about the module. Attribute one contains the present location of the module. When the module needs to be moved, an AGV will be sent to that location. The AGV will then take the module to the destination listed in either attribute two or three. When a module is ready to be moved, it must locate an empty stand in the area where it is to be taken. When an empty stand is found, its status is changed from empty to full by seizing the stand's resource. The number of the destination resource is stored in attribute eight of the module. This

prevents any other module from occupying the same stand before the module gets there. Then, a request for a vehicle is sent out. The signal is sent by a VWAIT node. The VWAIT node tells the system to send an AGV to the location listed in attribute or "label" one of the module. The idle AGV that is closest to the module will then proceed to the requesting module's location. When the AGV arrives, the module is loaded within 30 seconds and just upon departure, the module frees the location resource listed in attribute seven. Freeing the location resource signals that the stand is now empty. When the AGV departs to the destination listed in attribute two or three, attribute nine is assigned the current time. Upon arrival at the new destination, attribute nine is used to measure the time the trip took and the AGV is unloaded in 30 seconds. Finally, the AGV is then released to either travel to a parking loop or a new assignment.

#### *Receiving*

The receiving area consists of 6 workstations each with two pickup and delivery stands. The location of each stand is a control point in the network. The status of a stand, empty or full, is modeled by a resource. Each resource is identified by a number, 240 through 251; and a label (REC 1 through REC 12). Also, the resources are equivalenced to specific control points where the pickup and delivery stands are located. A resource in use indicates that the stand is full.

The receiving area is used to fill empty modules with stow items. At the start of the simulation, each stand contains an empty module ready to be filled. Loads for the module are generated at the rate of one every 315 seconds plus or minus 10 percent. Eighty loads are produced during the first seven hours of an eight hour shift if pick and stow runs are used in the high rise. If separate stow runs are made in the highrise, 60 loads are produced at the rate of one every 420 seconds plus or minus 10 percent. When a load is produced event one, a fortran subroutine, chooses which workstation the load will go to. This is done by randomly choosing an integer from one to six, each number representing a workstation. If the workstation has two loads already waiting for empty modules or was chosen to receive the previous load, another number is chosen.

A load cannot be released from a workstation unless there is an empty module present at at least one of the two pickup delivery stands assigned to that station. When that happens, the empty module is considered to be a stow module ready to be taken away to the module storage area for temporary storage until it is time to be taken to the highrise to have its contents stowed.

Before the stow module can be taken away, event two must be called to find a stand in the module storage area that the stow module can be taken to. The routine sequentially checks the status of the pickup and delivery stands in the module storage area pickup and delivery

stands. When one is found to be available (i.e. not in use), its status is changed to full, attribute two of the module is assigned the control number of the destination stand and attribute eight is assigned the status resource of the destination stand. Also, upon finding an empty stand, the stow module is assigned its row destination in the highrise (a randomly chosen integer between 1 and 60) and the number of stows it carries. The number of stows loaded into a module depends on how the highrise is being operated. If a pick and stow run is being made, one module shelf is filled with 20 to 30 items or if a stow only run is made, two shelves are filled with 30 to 40 items. Finally, an order is placed for an AGV to pick up the stow module.

When the AGV arrives, the module is loaded in 30 seconds and another order is made for an AGV to bring an empty module from the module storage area to the pickup delivery stand that the stow module is vacating.

#### *Module Storage Area*

The module storage area consists of 74 pickup and delivery stands. The status of each stand, empty or full is indicated by a resource (numbered 1 through 74 and labeled MSA 1 through MSA 74) which is equivalenced to the control point location of each stand. When a resource is in use, the stand is full. At the beginning of the simulation, each of the 74 stands contain an empty module. Stow modules come to this area from the receiving area to be sent later to the highrise. Empty modules are sent from the module storage

area to the receiving and highrise areas. The consolidation and highrise area sends empty modules to the modules storage area.

Module stands 73 and 74 are used to determine if more stands are needed in the module storage area. If a stow module in receiving cannot find an available stand in the module storage area, it is sent to stand 73. Stand 73 has a capacity to hold 100 modules (for simulation purposes). Stand 74 serves the same purpose for empty modules being sent to the module storage area. Every fifteen minutes, the total number of modules in the module storage area are counted. If the number is greater than 74, then more stands may be needed to be built.

#### *Highrise Rack and Bin Storage Area*

The Highrise contains 60 rows, each with a stand for deliveries only, a stand for pickups only, and an automated storage and retrieval machine. At the beginning of the simulation, each row is initialized with one empty module in the delivery stand. Also, the total number of IPG (issue priority group) I, II and III picks are determined on the basis of three random normal distributions: IPG I, with mean 34 and standard deviation 7; IPG II, with mean 67 and standard deviation 10; and IPG III with mean 158 and standard deviation 16. The number of picks are rounded to the nearest integer. The average number of picks from all 60 rows is 15,540.



The number of picks to be placed in a module depends on whether one shelf (a pick and stow run) or two shelves (pick only run) are to be filled. A single shelf is filled with 20 to 30 items and two shelves are filled with 40 to 60 items. Ninety percent of the time it takes 66 seconds to pick an item, the other 10 percent of the time it takes 107 seconds. Stowing an item takes 115 seconds 91.5 percent of the time and 188 seconds the remaining 8.5 percent of the time. When a combined pick and stow run is made, the total time of the run is the sum of the individual pick and stow times.

Two different types of runs may be simulated. The first is a combined pick and stow run where all the IPG Is are picked first, then the IPG IIs are picked. When there are no remaining IPG Is or IIs, and if a stow module destined for that row is in the module storage area, a combined pick of IPG IIIs and stow run is made. If a stow module does not exist, IPG IIIs are picked into an empty module. The second run type is a variation of the first type. The first method picks all the IPG Is first into an empty module. That module is then sent immediately to the consolidation area, even if there is still room in the module for additional picks. The second method fills the module to capacity with IPG IIs.

All the empty modules waiting in delivery stands are sent to a special AWAIT node to wait for the resource representing the ASR machine located in the row it has been

delivered to. When a crane resource is freed, the AWAIT node calls an ALLOC subroutine. The ALLOC subroutine checks all the modules (entities) to see if one is waiting for that particular resource. If there is, the module is released and the crane resource is seized to show it is in operation.

The released module calls the event five subroutine to determine how many picks to make, what types of picks (IPG I, II or III), how to pick, how many stows to make, and the total time the ASR machine will spend making picks and stows. The time an ASR machine spends making picks and stows is the sum of the individual pick and stow times. If there is not an order for an AGV to bring an empty module to the delivery stand, then one is made.

To simulate the crane, an activity with a duration time of the total picking and stowing time plus 40 seconds for loading/unloading the module is used. When a crane is finished picking a module, it cannot set the pick module down into the pickup stand until an empty module is delivered. A module is set down when the empty module ordered is delivered or if the empty crane does not have to make any more picks or stows. When the empty module is delivered, the pick module is set down in the pickup stand and the empty module is picked up by the crane to begin picking or stowing. Before the pick module can be taken away, it must first find a workstand or a buffer stand in the consolidation area. If all the workstation stands are full and a buffer stand is empty, the resource for the

buffer stand is seized and an order for an AGV is made. If a buffer stand is not available, the module waits in the pickup stand until one is available. When the module leaves for the consolidation area, an order is placed for an empty module to be brought to the delivery stand if it is needed.

If a stow only run is being made at the end of the day, the empty module at the end of the run is left in the delivery stand instead of being sent to the module storage area. This measure helps relieve the congestion of empty modules in the module storage area at the end of a shift.

#### *Consolidation Area*

The consolidation area is composed of 32 workstations, each with one pickup and delivery stand and 10 buffer pickup and delivery stands. Each workstation and buffer stand is assigned a resource to indicate if it is in use or not in use. When a pick module is delivered to a workstation, the AGV is unloaded by the module in 30 seconds then the module is unloaded at the rate of 30 seconds per pick item. When the module is emptied, a call is made for an AGV to take the module away. If a row in the highrise needs an empty module, then it is taken there. Else, the empty module is sent to the module storage area.

The buffer stands are used if all the work stands are full. Pick modules coming to the consolidation area will go to the buffer stand if the workstand is not available. When a workstand is freed, its resource is immediately seized by a pick module in the buffer. A call

for an AGV is made to take the pick module from the buffer to the workstand.

### *Pallets*

Pallets, representing a single item or a group of identical items too large to fit in the highrise, are created at a rate of one every 271.7 seconds for the first four hours of a shift and one every 469.6 seconds for the following three hours. In the IMC, a pallet is unloaded from a truck and placed into the system. A total of 76 pallets are created in 7 hours. A pallet is placed on one of six LTL (less than a truckload) pickup and delivery stands to be taken to the pallet storage area or the 3 P&M (preservation, packing, packaging, and marking) area. At the LTL stand a pallet is placed on is determined by a cyclical draw (i.e. 1, 2, 3, 4, 5, 6, 1...). A pallet has a 5 percent chance of needing additional work at the 3 P&M stands and a 95 percent chance of being sent to the pallet storage area. When a pallet is placed on an LTL stand, a call is made for an AGV to take it to the 3 P&M area or the pallet storage area.

If the pallet goes to the pallet storage area, a random integer between 1 and 29 is drawn to see which row in the pallet storage area the pallet is destined for. When a pallet is delivered to the pallet storage area, the pallet entity is terminated. No attempt was made to simulate the stacking of the pallet in the aisle by a fork lift.

A pallet going to the 3 P&M for additional work is taken to one of the nine 3 P&M stands. The AGV is released and the pallet undergoes the 3 P&M process for 1322 seconds plus or minus 10 percent. After that, the pallet is taken to the pallet storage area. Occasionally, a pallet at the 3 P&M stands must be sent back to the LTL to be removed from the system. The pallet may be damaged beyond repair, or the shipment refused because of a contractual dispute. This happens approximately once every eight days, a number so small that it was not simulated. Once each hour for the first four hours, a pallet is created in the pallet storage area to be taken to the pallet shipping stand in the consolidation area. Each pallet is randomly assigned a row number between 1 and 29 where the AGV comes to pick them up and take them to the shipping stand. Once at the shipping stand the AGV is freed and the pallet terminated.

#### *Tandem and AGV Computers*

In the IMC, whenever an AGV is needed, a request is sent to the Tandem computer. The Tandem passes the request on to the AGV computer for execution. In the simulation, a stand requesting an AGV places an order (entity) into the order queue (ORDQ). The order queue acts as the Tandem computer. The AGV computer or "order processor" receives its commands from the order queue. The order processor (event 11 in the fortran code) examines every order in the queue every 10 seconds (fig 24). An order is examined to

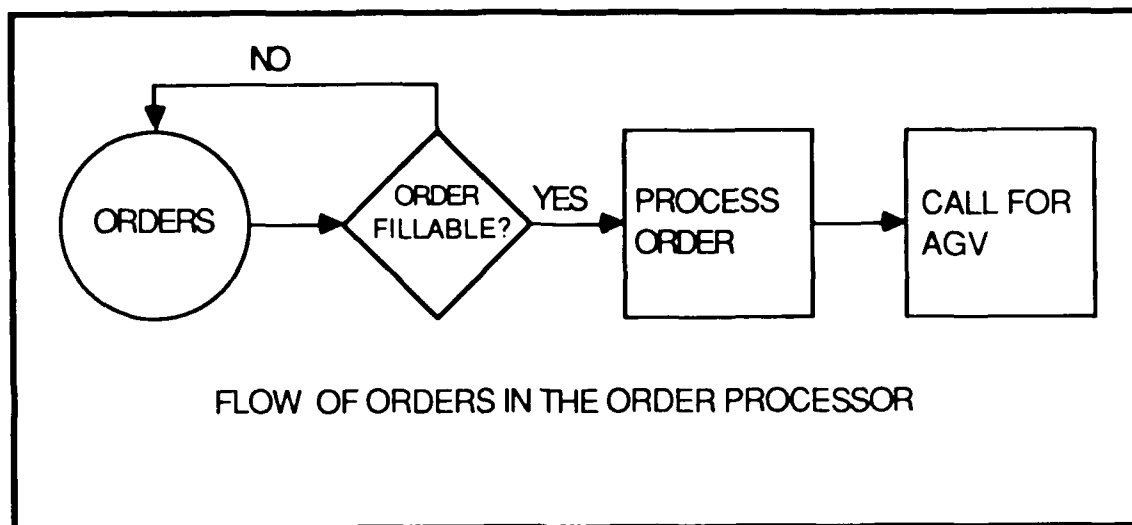


Figure 24. Order Processor

determined if it can or cannot be filled. An order tells the order processor where a vehicle is needed and what type of action should be taken (table 2). All orders are handled

Order Type	Action
1	Pickup stow module in receiving and take to the module storage area
2	Bring empty module from the module storage area to receiving
3	Bring empty or stow module to the highrise
4	Take pick module from the highrise to the consolidation area
6	Take pick module from buffer to the consolidation area
7	Take empty module from the highrise to the module storage area
8	Pickup a pallet and deliver

Table 2. Order types

on a first come, first served basis. However, if an order is unable to be filled, it is put back in the order queue to be evaluated 10 seconds later during the next cycle.

Order types one and two (fig. 25), deal with the movement of stow modules from the receiving area to the module storage area and the flow of empties in the reverse direction. An order of type three is the most complex type of order to fill. When an ASR machine in the highrise requests an empty or stow module be brought to that row, the computer first checks the consolidation area to locate an empty module. When an empty module is found, an AGV is dispatched to bring it to the highrise row. If an empty module is not found, the procedure is repeated in the module storage area. The order is refiled if an empty module is not found in either area. A request for a stow module causes the computer to check all the stow modules in the module storage area for one that is destined for that row. The stow module is then sent to the highrise. However, if a stow module is not found, and there are still picks to be made in that row, an empty module is sent instead. Order types four and six are used to transport pick modules to and around the consolidation area. Order type seven takes empty modules from the highrise to the module storage area near the end of a shift. Finally, order type eight calls AGVs to move pallets around the IMC.

A feasible order is placed in a VWAIT node. The VWAIT node signals the SLAM event calendar to send an AGV to the

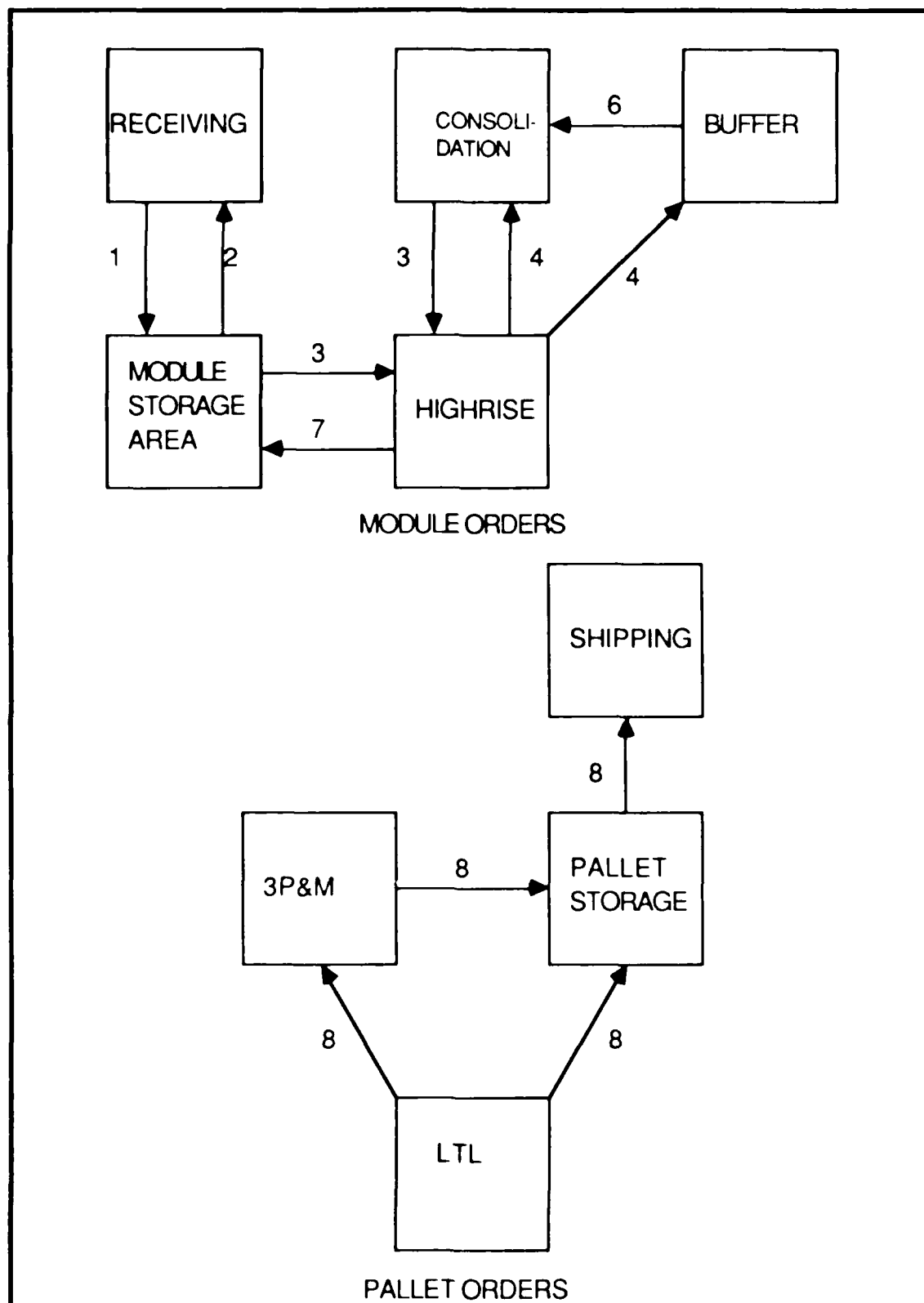


Figure 25. Flow of Orders



location listed in the order and take it to the destination also listed in the order.

The order processor does not tell the AGV which route to take. The AGV determines the shortest path between two points on its own. The path does not account for traffic congestion.

#### *Events Not Simulated*

During a normal eight hour shift, about 18 out of 25 vehicles will require a battery change. If the vehicles are not operating at full capacity, then there will be enough available time during a shift for a vehicle to be taken off line to change its battery. Thus it would have little impact on the simulation.

All vehicles and other pieces of equipment are assumed to function at a 100 percent reliability rate. Data concerning the reliability of the systems was not available to be included in the simulation.

#### *Summary*

This chapter along with chapter III, have presented a general overview of the workings of the IMC and how they are simulated using a computer simulation language. If further information is needed on the exact movement and flow of work in the IMC, a complete copy of the computer source codes for the simulation can be found in appendixes. Chapter V will present the results of the simulation discussed in this chapter.

## V. Results

### Introduction

To gather the necessary data to evaluate the performance of the IMC, a total of eight simulation runs were conducted. The primary measure each run was compared against was whether or not 15,500 picks were transported from the highrise and processed in the consolidation area. Each run consisted of eight to ten replications. Table three shows the settings and the number of replications in each simulation run.

Run #	Replications	# of AGVs	Pick Mode
1	10	20	A
2	10	25	A
3	10	30	A
4	8	35	A
5	10	20	B
6	10	25	B
7	10	30	B
8	9	35	B

Table 3. Simulation Run Descriptions

The size of the AGV fleet was varied between 20 and 35 by increments of 5. Two different modes of picking items in the highrise were used. Mode A represented the picking of all the IPG I items first and then picking the IPG II items second. Finally, a combined pick and stow run is made to pick the IPG III items and stow items from receiving. Mode B is the same as Mode A, except that if there is room left in a pick module containing IPG I items, the module is filled to capacity with IPG II items.

Each replication ran for 28,800 seconds of simulation time to represent an eight hour shift. The IMC only operates eight hours a day. At the beginning of each replication, the IMC is in a "reset" state. All the module storage stands, highrise delivery stands, and module loading stands are filled with empty modules. The system is "reset" after the eight hour morning shift after the shift is over. The simulation is reset by the INTLC subroutine.

Runs four and eight have fewer replications due to traffic jams caused by the AGVs. In a real situation, the traffic jam could be unsnarled by nearby workers. However, in the simulation, it would have been extremely difficult to model the workers actions in unsnarling the traffic jam. Thus, with no practical means to stop the jams, the results of the replication were severely distorted. The jams always occurred near the buffer stands in the consolidation area at approximately five hours of simulation time. The jam prevented the AGVs from bringing anything in or out of the consolidation area or from performing work anywhere else in the IMC.

#### *Consolidation Pick Processing*

*Performance Goal.* The average number of picks processed by the consolidation area workstations (table four) was calculated from the replications of each run. The total number of picks processed by the consolidation area workstations was compared against a performance measure

# Picks Processed

Run #	Mean	Standard Deviation	Test Statistic	p-value
1	15413	174	-1.58	.07
2	15529	149	.61	.7
3	15476	176	-.04	.5
4	15516	150	.33	.6
5	15447	155	-1.08	.16
6	15534	150	.72	.8
7	15480	124	-.51	.3
8	15535	162	.68	.7

Table 4. Total Picks Processed by the Consolidation Workstations

(15,500 picks processed in the consolidation area) for the IMC. A test of the means (one tailed t-test) was conducted with a null hypothesis that the average number of picks processed in a run was greater than or equal to the population mean of 15,500 picks (the performance goal). The test statistic and the p-values for the tests are also found in table four. The p-values greater than .16 are rough estimates interpolated from a t distribution table. At an alpha level of .05 or smaller, all the runs are considered to be insignificant. Thus, the null hypothesis should not be rejected and all the runs should be considered to have met or exceeded the 15,500 pick goal.

*Effect of AGVs and Pick Mode.* The purpose of varying the AGV fleet size and the pick mode was to determine their effect on the number of picks processed in the consolidation area. An unbalanced ANOVA test was used to evaluate these effects. Two separate null hypotheses were tested. The

first was that the AGV fleet size had a significant effect on the number of picks processed. The second hypothesis was that the pick mode had a significant effect on the number of picks processed in the consolidation area. The results of the test are in table five. Both the AGV fleet size and the

Source	Degrees of Freedom	Sum of Squares	Mean Square	F Value	p Value
AGV Fleet	3	131,342	43,781	1.88	.14
Pick Mode	1	4,536	4,536	.20	.66
Error	72	1,672,766	23,233	-	-
Total	76	1,808,645	-	1.46	.22

Table 5. Results of Variance Test on Effects of AGV Fleet Size and Pick Mode

pick mode did not have any significant effect upon the number of picks processed in the consolidation area if an alpha value of less than .14 is used.

The results of this test are only valid in the context of the number of picks available in the highrise to take to the consolidation area. In this simulation, that number is 15,550 picks available in the highrise.

*Consolidation Workstation Workload.* The number of picks processed by a consolidation workstation varies from station to station. A large variance in the workload would indicate that some of the 32 workstations are getting more work than the others. Data was collected to determine the average number of picks processed by each station (table 10). The average number of picks processed by a

Run #	Mean	Standard Deviation
1	482	36
2	485	54
3	487	63
4	484	62
5	482	35
6	485	48
7	484	57
8	483	60

Table 6. Consolidation Workstation Workload

workstation was about the same for all the runs. However, as the vehicle fleet size increased, the variance increased. The increase means the distribution of work between workstations is becoming more unbalanced.

#### *AGV Efficiency*

*Measuring Efficiency.* The efficiency of the AGV fleet was defined as the total percentage of the time during a replication that the AGV fleet was either carrying a load, traveling to pickup a load, loading a module, or unloading a module. Efficiency was measured in two ways, interval and cumulative.

The first method measured the interval AGV fleet efficiency over a fifteen minute interval. Figures 26 and 27 plot the fleet interval efficiency for all four vehicle fleets, given pick mode A or B. The two figures are very similar. The AGV fleet with 20 vehicles operates at a higher efficiency rate than the other sized fleets at any given interval because there are enough jobs to keep the fleet busy. Three specific area show up in these figures

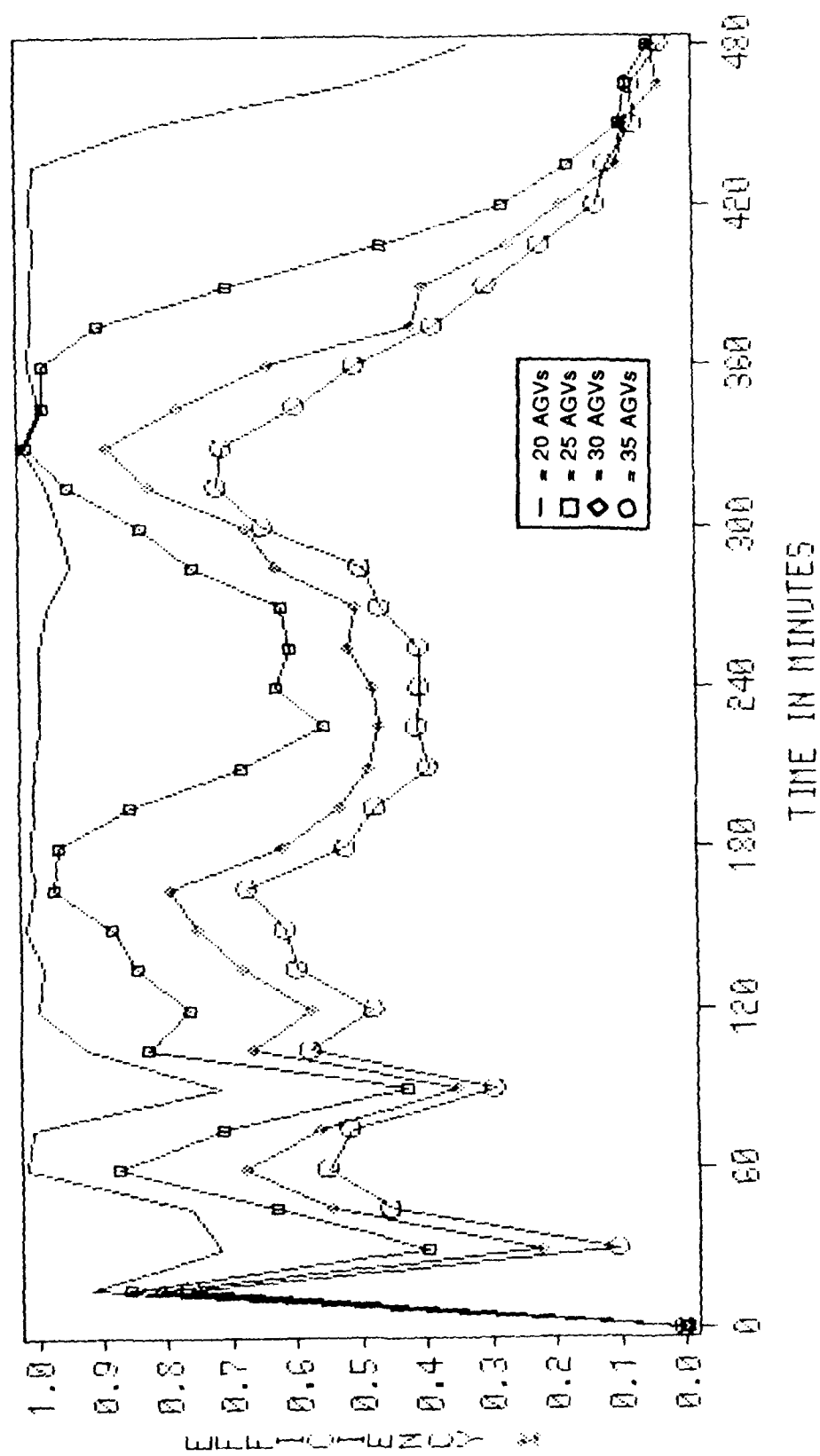


Figure 26. Interval AGV Fleet Efficiency - Pick Mode A

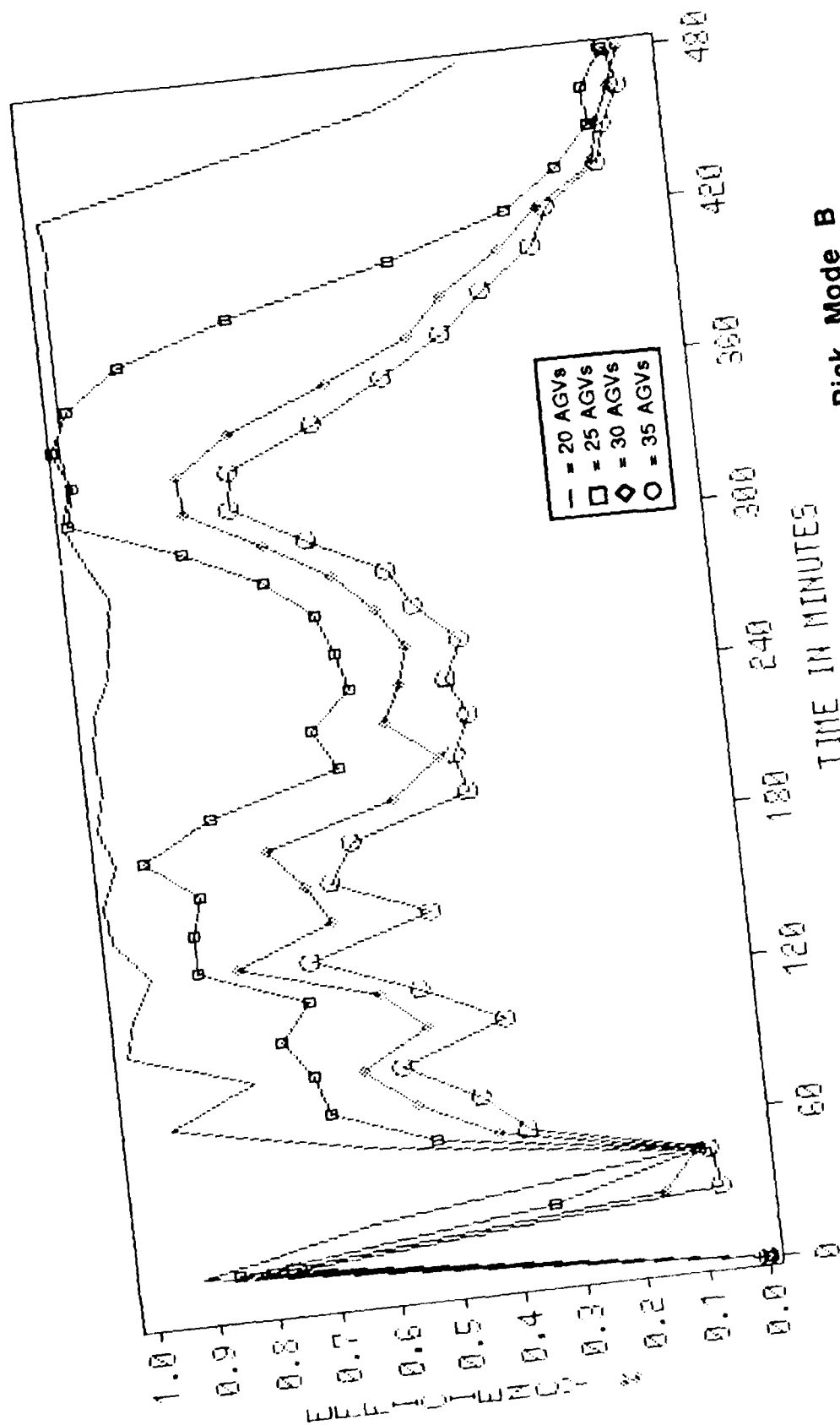


Figure 27. Interval AGV Fleet Efficiency - Pick Mode B



and the rest to follow. The first was between 0 and 90 minutes and represents the AGVs taking modules to the highrise and then taking the IPG I picks to the consolidation area. The second period was from 91 minutes to 240 minutes. The efficiencies have dropped while the ASR machines stop picking IPG I items and start to pick IPG II items. Then the activity starts to pick up as more IPG II items are taken to consolidation area. The activity tapers off as the last of the IPG II items are picked. The third period went from 241 to 480 minutes. Activity increases sharply because of the large number of IPG III items to pick. This number translates into a high demand for AGVs. As the number of IPG III items left to pick decreases, there is little work left for the AGVs and the efficiencies decrease. However, the 20 vehicle fleet does not get its work done as fast as the larger fleets. So, it ends up working longer.

The second method for measuring efficiency was to take a cumulative average over the entire simulation. The cumulative average was sampled every 15 minutes. The last sample taken at 480 minutes is the overall fleet efficiency average for the entire run.

Figures 28 and 29 show the cumulative AGV fleet efficiency sample for each vehicle fleet given one of two pick modes. The plots for the two pick modes are very similar. The fluctuations in the averages were due to the three time periods mentioned earlier. The 20 vehicle fleet

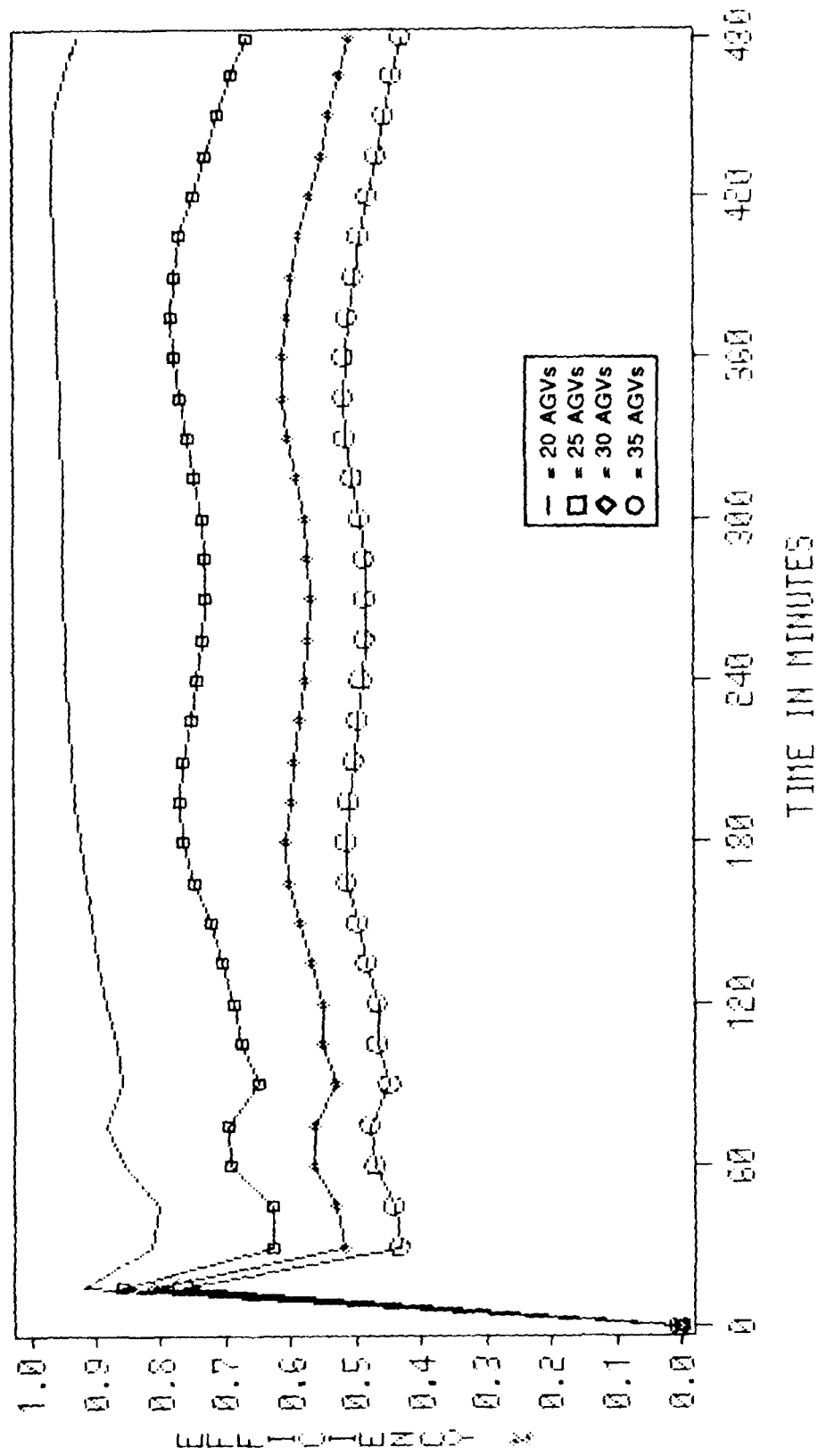
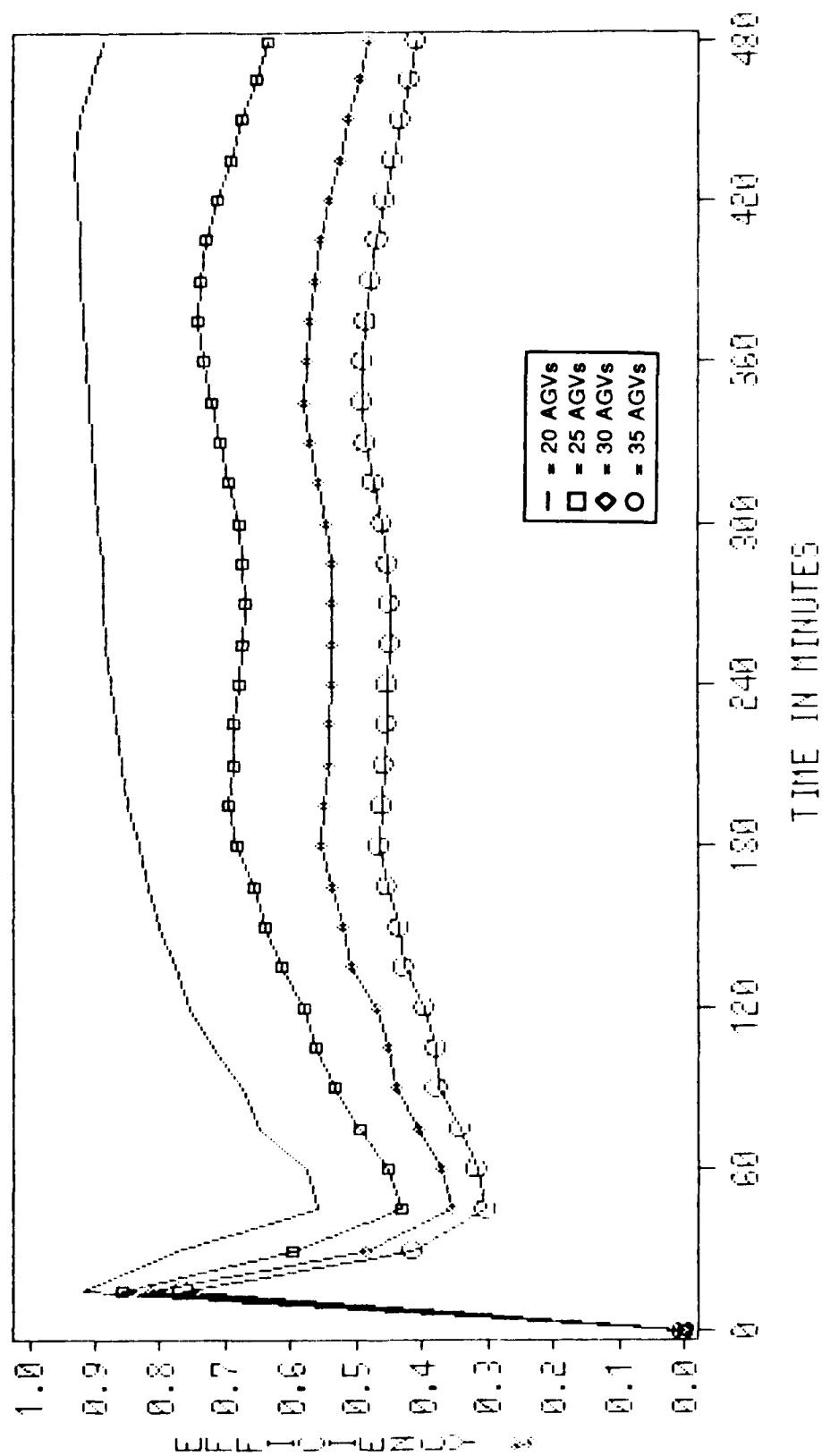


Figure 28. Cumulative AGV Fleet Efficiency  
- Pick Mode A



**Figure 29. Cumulative AGV Fleet Efficiency**  
- Pick Mode B

has the highest cumulative efficiency of the four fleets because it is doing the same job with less vehicles.

*Effect of Pick Mode.* The two pick modes used had little effect on the efficiency of the AGV fleets. Figures 30 and 31 compare the two pick modes by contrasting the interval and cumulative efficiencies of a 25 vehicle fleet. In pick mode A, the first group of modules left the highrise approximately 15 minutes earlier than the first group of modules from pick mode B. In pick mode B, the ASR machine spends the extra time picking additional IPG II items. So, while the ASR machines are picking the extra items, the AGVs did not have any jobs to perform. As a result, the interval efficiency (30 to 45 minutes) drops sharply (figure 30) and the overall cumulative efficiency at 45 minutes (figure 31) shows a corresponding drop. After 120 minutes, the curves for both pick modes are almost the same.

#### *AGV Routing*

AGVs can make one of thirteen possible routes to deliver a pallet or module (table seven). Data was collected to evaluate the effect that the vehicle fleet size and the pick mode had on the travel time and number of trips along a given route. The average number of trips made and average trip times for each route are shown in tables eight and nine. The data from run 4, 35 vehicles and pick mode A, was not included, because the results were lost in a hard disk failure in the computer system.

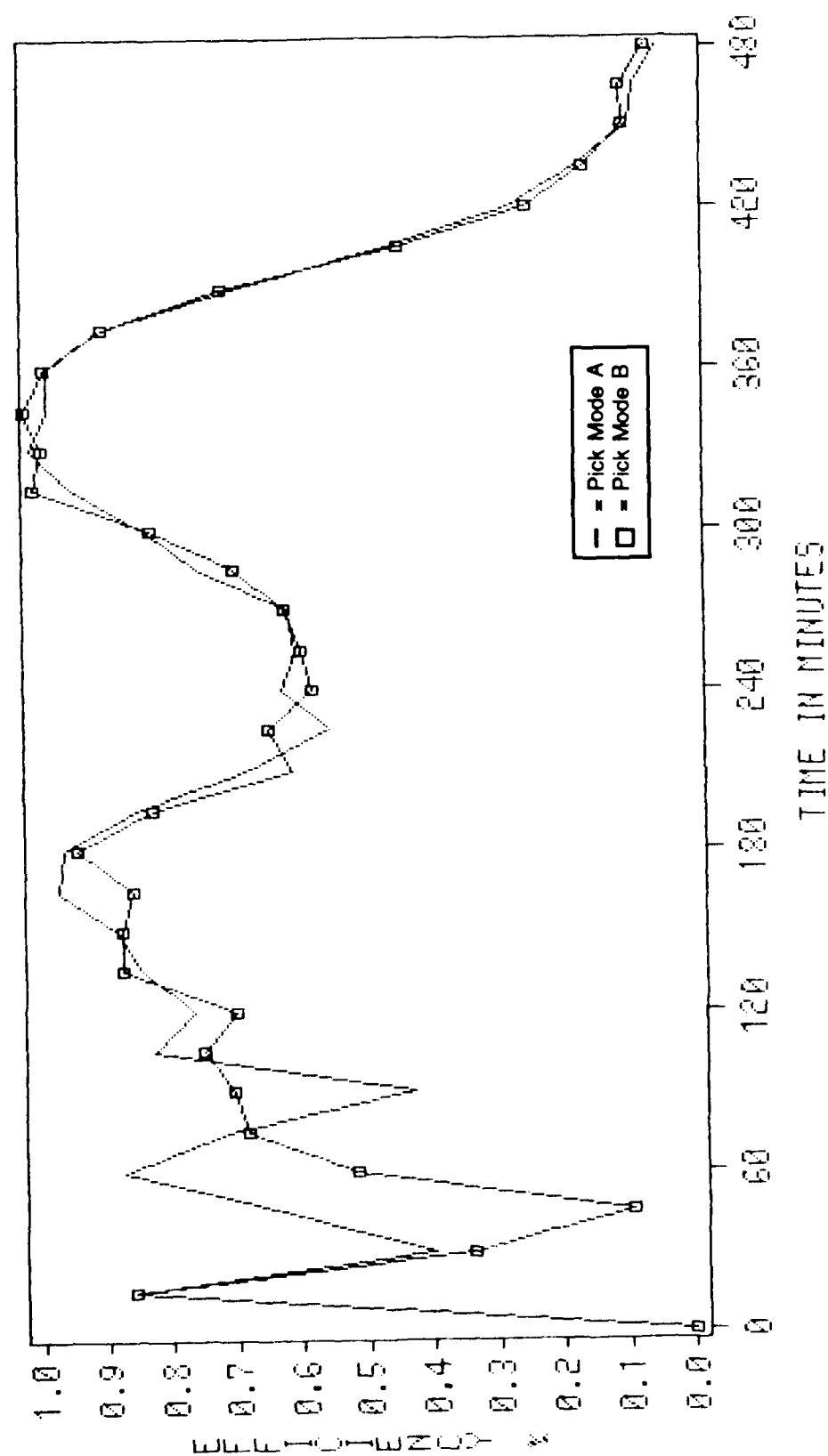
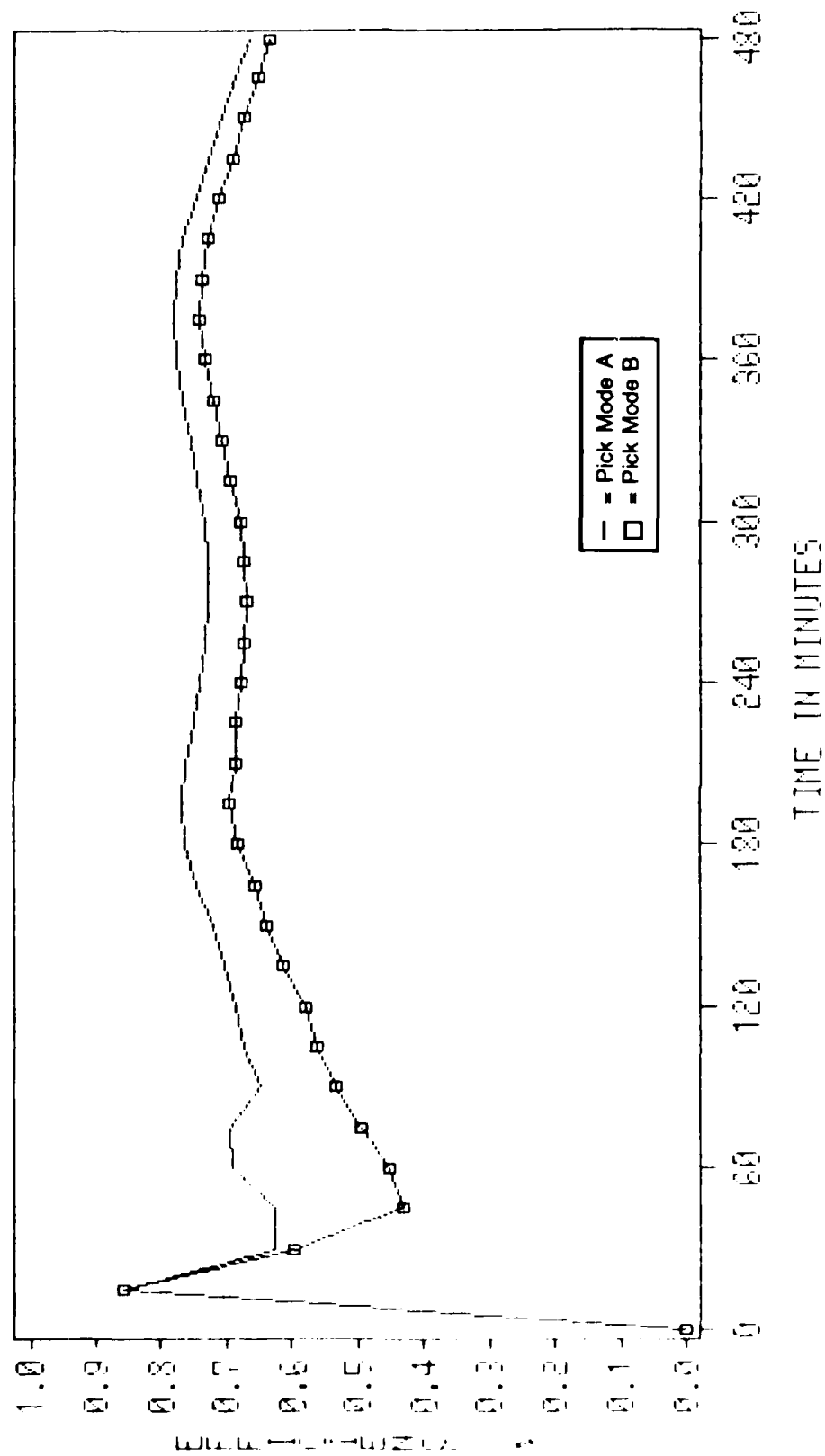


Figure 30. Interval AGV Fleet Efficiency - 25 AGVs



**Figure 31. Cumulative AGV Fleet Efficiency - 25 AGVs**

Route #	From	To	Cargo
1	Receiving	Module Storage	Stow Module
2	Module Storage	Receiving	Empty Module
3	Highrise	Buffer	Pick Module
4	Highrise	Consolidation	Pick Module
5	Buffer	Consolidation	Pick Module
6	Module Storage	Highrise	Empty/Stow Module
7	Consolidation	Highrise	Empty Module
8	Consolidation	Module Storage	Empty Module
9	Highrise	Module Storage	Empty Module
10	LTL	Pallet Storage	Pallet
11	LTL	3 P&M	Pallet
12	3 P&M	Pallet Storage	Pallet
13	Pallet Storage	Pallet Shipping	Pallet

Table 7. Vehicle Delivery Routes

RUN							
Route	1	2	3	5	6	7	8
1	80	80	80	80	80	80	80
2	80	80	80	80	80	80	80
3	216	193	178	222	196	178	192
4	190	216	231	155	185	203	191
5	216	193	178	222	196	178	192
6	150	149	149	148	145	145	145
7	248	250	250	220	223	222	225
8	151	156	156	151	156	153	154
9	7	12	11	8	11	10	11
10	72	71	72	72	72	72	71
11	4	5	5	4	4	5	5
12	4	5	5	4	4	5	5
13	4	4	4	4	4	4	4

Table 8. Number of AGV Trips Over a Given Route

Unbalanced ANOVA tests were used to evaluate the effects of the fleet size, route, and pick mode on the number of trips made on a route (table 10) and on the time to make a trip on a route (table 11).

In the first test, three hypothesis were used. The first was that the vehicle fleet size had a significant effect on the number of trips made on a route. The other

Route	RUN						
	1	2	3	5	6	7	8
1	35	36	39	39	39	39	43
2	48	49	49	49	48	49	50
3	231	231	232	232	232	231	231
4	248	248	249	249	247	253	264
5	66	66	69	64	68	68	80
6	168	228	289	166	232	288	334
7	201	203	212	198	200	209	227
8	232	267	299	236	285	317	407
9	168	278	286	166	232	288	334
10	289	290	333	289	293	329	430
11	138	151	180	139	147	182	154
12	391	388	403	383	385	410	458
13	217	215	215	212	212	214	214

Table 9. Average Time Along a Given Route

two hypothesis were similar to the first except that they stated that the route and the pick modes had an effect on the number of trips made on a route. The results showed that only the route had a significant effect on the number of trips made on a route (p-value = .0001). This was not surprising because certain routes have less traffic (table seven). The pick mode had a very slight effect on the number of trips made on a route (p-value = .058). The number of AGVs had no effect on the number of trips on a route (p-value = .31).

Source	Degrees of Freedom	Sum of Squares	Mean Square	F Value	p Value
AGV Fleet	3	385	128	1.21	.3128
Route	12	612,449	51,037	480.76	<.0001
Pick Mode	1	393	393	3.70	.0583
Error	74	7,856	106	-	-
Total	90	621,083	-	361.03	<.0001

Table 10. Results of Variance Test on Route Trips



The same type of test was performed on the data in table nine to evaluate the effect of the same variables on the time to complete a trip. The results are found table

Source	Degrees of Freedom	Sum of Squares	Mean Square	F Value	p Value
AGV Fleet	3	43,799	14,600	13.17	.0001
Route	12	983,219	81,935	76.95	.0001
Pick Mode	1	540	540	.51	.4784
Error	74	78,789	1,065	-	-
Total	90	1,106,313	-	60.32	.0001

Table 11. Results of Variance Test on Trip Times

eleven. The type of route had a significant effect on the time to complete a trip (p-value = .0001). Each route is a different length, so the effect of length on travel time was expected to be significant. The AGV fleet size had a significant effect (p-value = .0001). The more AGVs in the system, the more delays an AGV encounters, which results in longer travel times. The pick mode did not have any significant effect (p-value = .4784). It was not expected to be significant since the pick mode only produces the modules and does not transport them through the system.

*Waiting Time for an AGV.* Figures 32 and 33 show the average number of modules awaiting an AGV for each pick mode. The data was sampled over fifteen minute periods. Each point represents the average number of modules awaiting transportation in that fifteen minute interval. The figures show how the smaller the AGV fleet size, the more orders are left waiting during an interval.

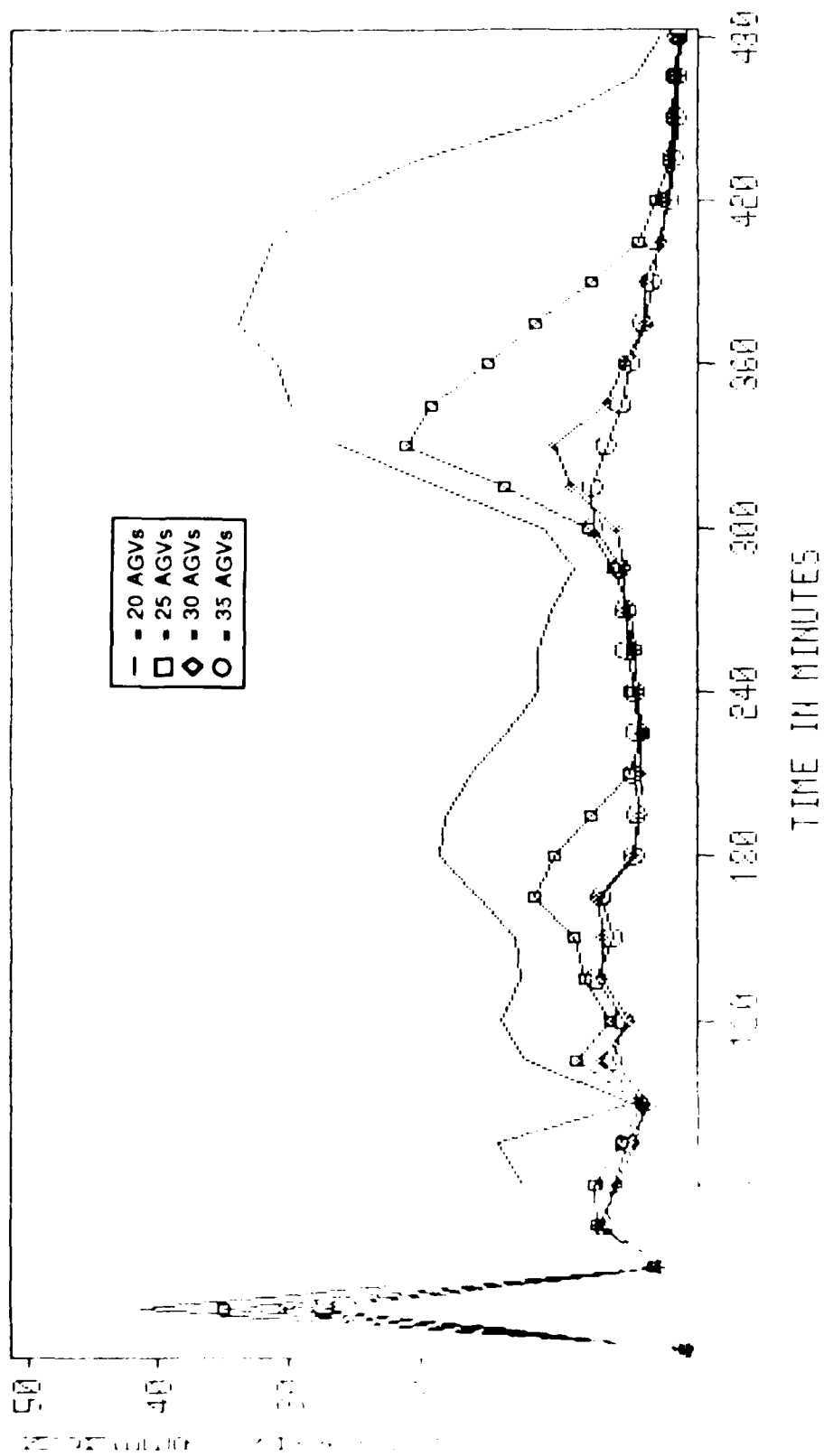


Figure 32. Number of Modules Awaiting Transportation  
- Pick Mode A

AO-A189 484

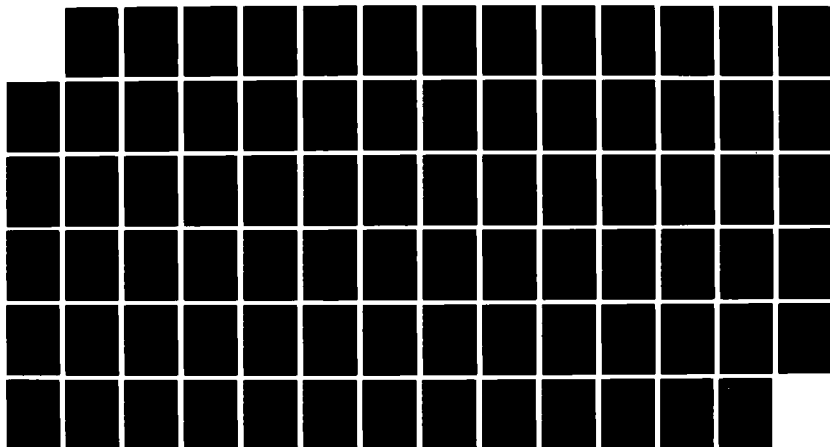
A SIMULATION OF THE DEFENSE LOGISTICS AGENCY'S  
INTEGRATED MATERIALS COMPLEX(U) AIR FORCE INST OF TECH  
WRIGHT-PATTERSON AFB OH SCHOOL OF ENGINEERING  
R B BERRY DEC 87 AFIT/GOR/ENS/87D-2

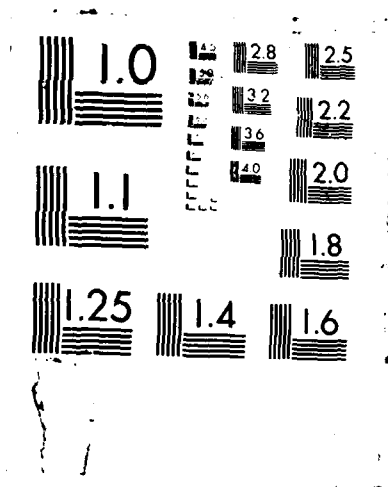
2/2

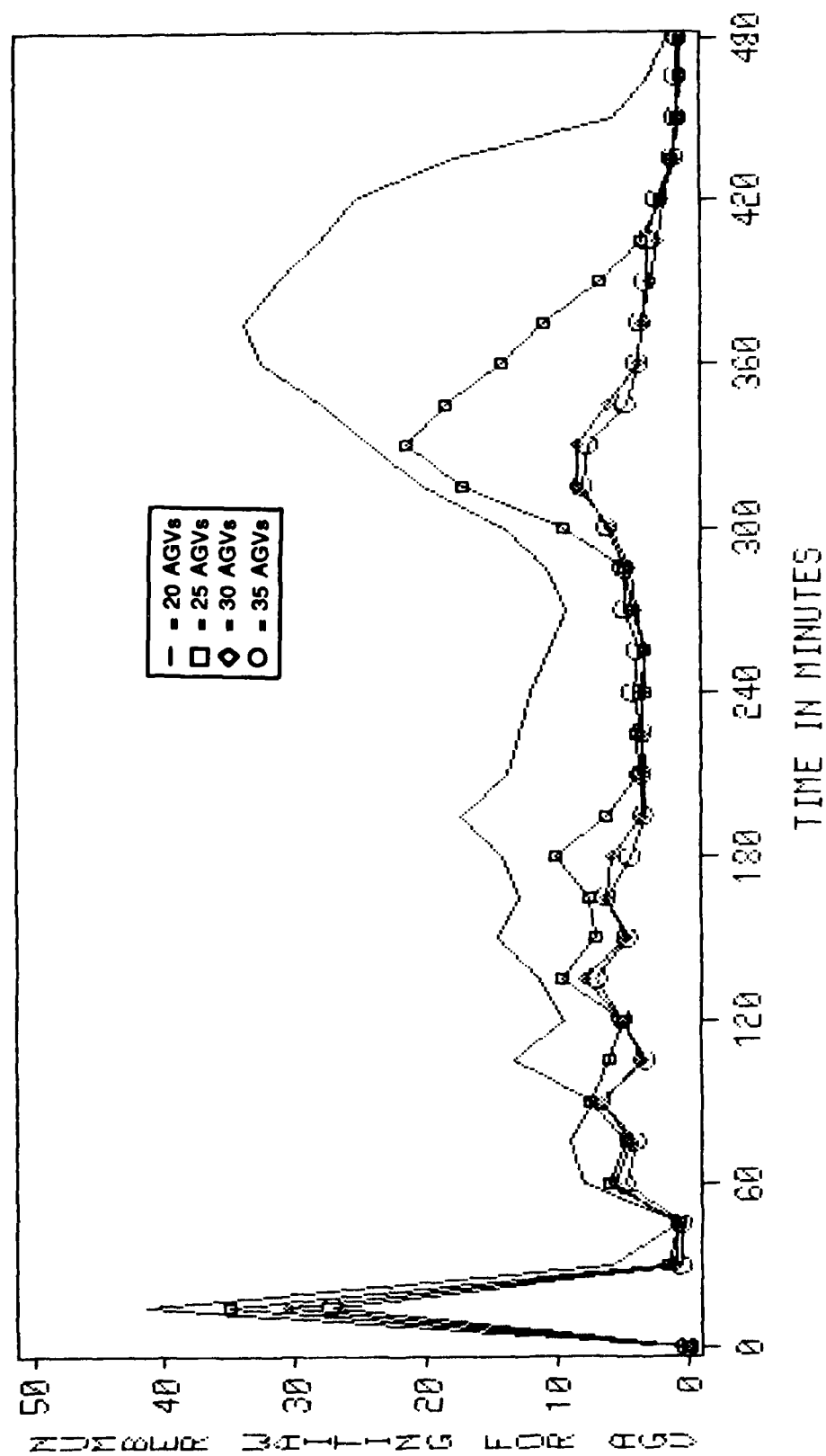
UNCLASSIFIED

F/G 15/5

ML







**Figure 33. Number of Modules Awaiting Transportation  
- Pick Mode B**

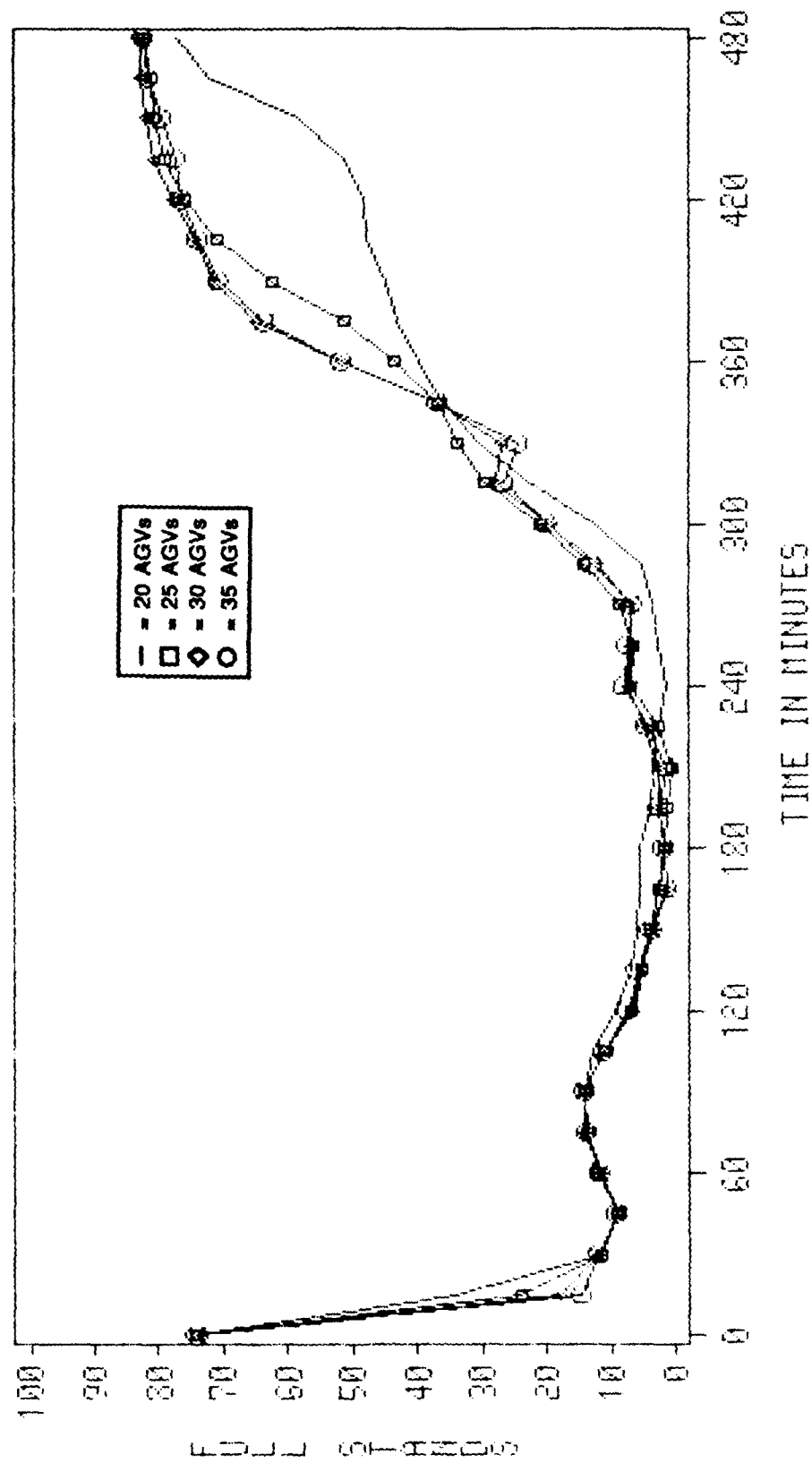
### *Module Storage Area*

To see if the number of stands in the module storage area was adequate, every fifteen minutes, the number of filled stands in the module storage area were counted. The total number of stands physically in the module storage area was 74. However, two stands, by means of the simulation, were allowed to hold more than one module. So, at any given time, there may be more than 74 modules in the module storage area. Figures 34 and 35 show the module storage area occupancy rate for each of the two pick modes. The number of modules in the module storage area increases past 74 to 85 near the end of the day when all the empties in the system are no longer needed.

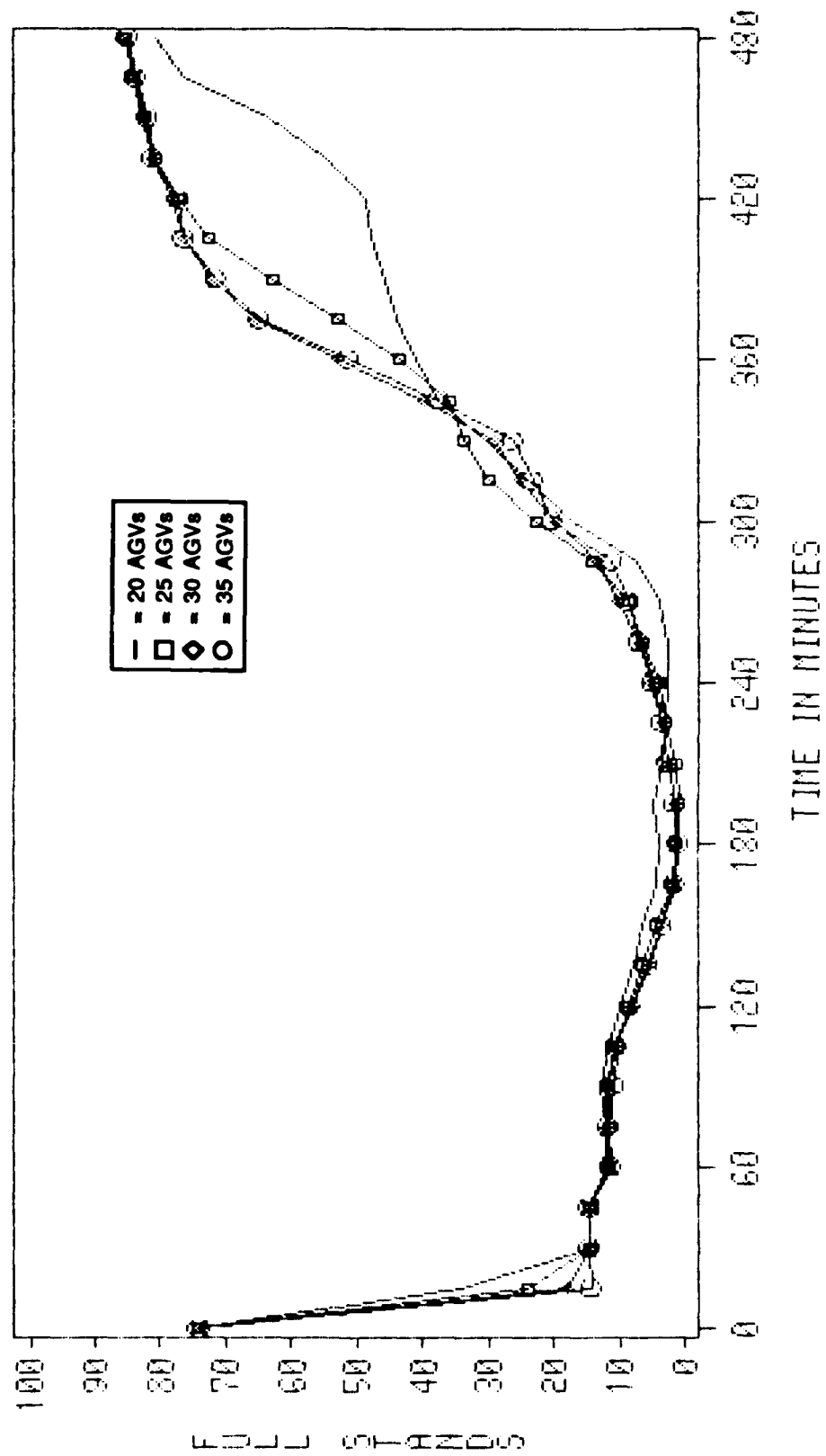
### *Commentary on SLAM*

Running such a large model using SLAM was not without some difficulties. The subroutine VRSETP contains two lines which in some circumstances fool the computer to think an AGV is still moving even though it is blocked by another vehicle. The modified subroutine is contained in appendix B.

Computation time increases sharply with an increase in the number of control points and segments. Computation is measured in CPU time, the total time the program uses the CPU in the computer. The first thing the program does is to compute a matrix with the shortest path from each control point to the other. For each route, dozens of shortest path calculations must be made. The matrix is only computed once



**Figure 34. Module Storage Area Occupancy Rate  
- Pick Mode A**



**Figure 35. Module Storage Area Occupancy Rate**  
- Pick Mode B



and used for all the replications in a run. There is no way to save the matrix for any additional runs. On a VAX 11/780, the computation time for the matrix using the segments of one IMC building takes 15 cpu minutes, two takes 25 CPU minutes, three takes 45 CPU minutes and four takes almost two CPU hours. On a newer VAX 8650 (an 11/780 with 4 CPUs), four buildings takes 35 CPU minutes. A run of 10 replications in the 8650 takes one hour and forty minutes of CPU time.

Computation time of the matrix is doubled if the control points and segments are numbered with large gaps between sets of points or segments. For example, if it can be avoided, do not number a set of segments from 1 to 100 and the next from 400 to 500.

The trip report function in the VCONT statement should be turned off for a large model. It creates a huge matrix with the dimensions of the highest numbered control point used. With this simulation, that matrix would be 604 by 604 elements. That large matrix causes the program to use up all available memory in the computer and slows down the processing of the simulation. The matrix, if printed out is 48,000 lines long, and takes up 4400 blocks of a VAX hard drive operating on VMS.

## Summary

The results exhibited in chapter V. focused on the area of the IMC performance goal, the consolidation area, AGV efficiency and travel, and the module storage. Chapter VI. will take these results offer some unique insights and recommendations about the aforementioned areas.

## *VI. Conclusions and Recommendations*

### *Conclusions*

The overall measure of performance for the IMC is to meet the goal of processing 15,500 picks from the highrise storage area in the consolidation area in one eight hour shift. The simulation showed that the average number of picks processed in each run met or exceeded the 15,500 pick goal (table four).

However, neither the size of the AGV fleet nor the picking mode had an effect upon the average number of picks processed in any of the runs. Using an alpha value of .14 or less would lead to the rejection of the hypothesis that the number of AGVs and the type of pick mode had a significant effect on the number of picks processed. This conclusion is based on a 15,500 pick goal. If the goal was set to a larger number of picks or a fleet of less than 20 vehicles is used, the AGV fleet size would start to have an effect on the performance goal.

The AGV fleet of 20 vehicles operated at a higher efficiency level in both pick modes than the other three vehicle fleets (figures. 26 through 29). Since it was operating at such a high efficiency rate, there was little room for improvement. Figures 32 and 33 show a larger number of orders waiting for an AGV from the 20 vehicle fleet. A high efficiency rate indicated that the AGV fleet had an extremely limited capacity to accept extra work and took longer to perform the work it had. If the number of

picks in the performance goal were to be continually increased, a point would be reached where the AGV fleet of size 20 would reach maximum efficiency. When this happens, the fleet would no longer be capable of handling the extra work and the performance goal would not be met. Thus, the size of the AGV fleet would have an effect on the number of picks processed.

Increasing the vehicle fleet size can be used as a method to meet an increased performance goal. The larger the vehicle fleet, the lower the efficiency rating. A lower efficiency rating indicated an unused potential to handle extra work since a larger percentage of the vehicles were idle. But, increasing the fleet size directly affects the time it takes an AGV to make a trip.

The larger the fleet size, the longer it takes an AGV to make a trip on certain key routes (table nine). The longer travel time is due to more congestion in the system. A test to evaluate the effects of the route, pick mode, and the vehicle fleet size on the time of a trip (table eleven) showed that the fleet size and route were significant factors at an alpha level of .0001. Picking mode did not have a significant effect on the trip times. Thus, a larger vehicle fleet may be able to handle an increased workload, but at a price of longer trip times caused by increased congestion. If the vehicle fleet were too large, the performance goal may not be met because of unacceptably long travel delays due to congestion.

Changing the pick mode caused the number of trips made on some segments to increase. In the model to evaluate the effects of route, pick mode, and AGV fleet size on the number of trips made on a certain route (tables eight and ten), only the route and the pick mode had a significant effect. A shift in the number of trips made on different routes could increase or decrease the total time an AGV fleet spends at work. Thus, a large vehicle fleet may perform slightly better at handling an increased workload using one pick mode than the other.

The average number of picks processed at a consolidation workstand (table six) was constant for each run. However, the standard deviation increased with the size of the AGV fleet. A consolidation stand is in use from the moment it is reserved by a module at a pickup stand in the highrise until the moment the same module leaves the consolidation stand. The time a stand is in use is much longer with a smaller vehicle fleet, because the time a module waits for an AGV at the highrise delivery stand and then at the consolidation stand is greater than with a larger fleet. Figures 32 and 33 show that the smaller the vehicle fleet, the larger the number of modules awaiting transportation and the longer the waiting time for an AGV.

When a module is looking for an available consolidation stand, it starts with stand one and stops when the first available stand is found. So, if a stand is tied up for a longer period, the module must look further down the list to

find an available stand. As a result, the stands are more evenly utilized and the standard deviation is lower.

The 74 pickup/delivery stands in the module storage were adequate enough to meet the demand until the last hour of the shift. Empty modules from the consolidation area and highrise caused the occupancy rate to increase to the mid 80s. This problem could easily be solved if a few modules were manually removed from the system by a forklift.

Traffic jams occurred (one in ten replications) in the runs of 30 and 35 AGVs using pick mode A. The jams occurred near the buffer stands in the consolidation. The cause of the jam was an AGV backing out of a spur and an approaching AGV on the segment the spur was connected to. Both AGVs prevented the other from moving, and a complete gridlock soon followed. The only suggested method for resolving a gridlock situation is for human intervention. Workers would have to separate the vehicles to reset the system.

The results of the simulation runs did not account for the reliability of the AGVs or any other system in the IMC. Hence, a fleet of 20 AGVs is more prone to be affected by the breakdown of one or more vehicles because it is already at the limits of efficiency. A larger fleet of vehicles would be able to meet the workload if a small number of vehicles broke down. However, if the breakdown causes a traffic jam, overall performance will suffer until the jam is unclogged.

## *Recommendations*

There are many areas in the simulation which would benefit from further sensitivity analysis. These areas include number of picks to be made in the highrise and additional pick modes. By increasing the number of picks made in the highrise, more AGV trips are required. The need for additional trips creates a demand for more AGVs. A smaller fleet may not be able to handle the demand.

Additional pick modes, where all the IPG III items are picked before the stow modules are taken to the highrise, should be tried. The expected results may show that the consolidation area will process all the picks much earlier in the day than the other modes. But, the ASK machines will remain busy making stows until the end of the shift. The module storage area occupancy rate will grow steadily because more stow modules will accumulate.

The interval AGV efficiency data (figures 26 and 27) showed how the efficiency of a large vehicle fleet fluctuated during a shift. Varying the number of AGVs in accordance with the demand or workload, could maximize the fleet efficiency and reduce congestion. Efficiency is increased because all AGVs in the system are working and with less AGVs, congestion is reduced. However, a plan to determine when to increase or decrease the numbers of AGVs in the system and where to hold the idle AGVs would need to be worked out.

This simulation is a perfect candidate to be translated into a graphic simulation language such as TESS or Cinema. The ability to graphically observe the movement of the AGVs would offer great insights into the causes of traffic jams and the utilization of certain segments.



## Appendix A: SLAM Simulation Code

```
GEN,RBERRY,AGV,9/22/87,10,N,N,Y,Y,Y/1;  
LIMITS,31,15,500;  
INIT,0,28802;  
CONT,0,1,1,2,N;  
VCONT,1,2,N,N,N;
```

```
:  
:   THESIS PROJECT FOR CAPT RICHARD B. BERRY, USAF.  
:   DEPARTMENT OF OPERATIONS RESEARCH  
:   AIR FORCE INSTITUTE OF TECHNOLOGY  
:   WPAFB, OH 45433  
:   3 DEC 1987
```

```
:  
:   THIS PROGRAM IS WRITTEN IN THE SLAM II SIMULATION  
:   LANGUAGE WITH THE MATERIALS HANDLING EXTENSIONS.  
:   IT IS DESIGNED TO SIMULATE THE MOVEMENT OF AUTOMATED  
:   GUIDED VEHICLES IN THE DEFENSE LOGISTICS AGENCY'S  
:   INTEGRATED MATERIAL COMPLEX.
```

```
:  
:   ARRAY 1 CONTAINS THE CONTROL POINTS FOR ALL THE MODULE  
:   STORAGE AREA PICKUP/DELIVERY (PD) STANDS #1 THRU #74.
```

```
:  
:   ARRAY(1,105)/23,24,26,27,28,30,31,32,33,35,36,37,39,40,41,42,44,  
:               45,46,72,73,74,75,77,78,79,81,82,83,84,86,87,88,90,91,  
:               92,93,95,48,49,50,51,53,54,55,57,58,59,60,62,63,64,  
:               66,67,68,69,96,97,99,100,101,102,104,105,106,108,109,  
:               110,111,113,114,115,117,118;
```

```
:  
:   CONTROL POINTS FOR HIGHRISE STORAGE ROWS #1 THRU #60
```

```
:  
:   ARRAY(2,105)/220,221,223,202,225,226,228,204,230,231,233,206,  
:               235,236,238,242,243,245,219,247,248,250,212,252,  
:               253,255,214,257,258,260,302,300,299,308,297,295,  
:               294,310,292,290,289,312,287,285,284,280,278,277,  
:               317,275,273,272,319,270,268,267,321,265,264,262;
```

```
:  
:   CONTROL POINTS FOR CONSOLIDATION STANDS #1 THRU #32  
:   AND CONSOLIDATION BUFFERS #1 THRU #10
```

```
:  
:   ARRAY(3,105)/447,405,448,406,450,408,451,409,453,411,454,412,  
:               456,414,457,415,459,417,460,418,462,420,463,421,  
:               465,423,466,424,468,426,469,427,  
:               430,431,433,434,436,437,439,440,442,443;
```

```
:  
:   CONTROL POINTS FOR MODULE LOADING PD STANDS #1 THRU 12
```

```
:  
:   ARRAY(4,105)/3,4,6,7,9,10,12,13,15,16,18,19;
```

```
:  
:   CONTROL POINTS FOR LTL PALLET STANDS
```

```
:  
:   ARRAY(5,105)/148,149,151,152,154,155;
```

:  
: CONTROL POINTS FOR 3P&M  
:  
: ARRAY(6,105)/124,125,127,128,130,131,133,134,136;  
:

: CONTROL POINTS FOR PALLET STORAGE  
:  
: ARRAY(7,105)/547,549,551,553,555,557,559,561,563,  
: 565,567,569,571,507,509,511,513,515,  
: 517,519,521,523,525,527,529,531,533,  
: 535,537;  
:

NETWORK;  
:

: EACH MSA RESOURCE REPRESENTS A MODULE STORAGE AREA PD  
: STAND. IF MNRSC = 0, THE STAND IS FULL, OR ABOUT TO BE  
: FILLED. IF MNRSC = 1, THE STAND IS EMPTY.  
:

RESOURCE/1,MSA1(1),27;  
RESOURCE/2,MSA2(1),27;  
RESOURCE/3,MSA3(1),27;  
RESOURCE/4,MSA4(1),27;  
RESOURCE/5,MSA5(1),27;  
RESOURCE/6,MSA6(1),27;  
RESOURCE/7,MSA7(1),27;  
RESOURCE/8,MSA8(1),27;  
RESOURCE/9,MSA9(1),27;  
RESOURCE/10,MSA10(1),27;  
RESOURCE/11,MSA11(1),27;  
RESOURCE/12,MSA12(1),27;  
RESOURCE/13,MSA13(1),27;  
RESOURCE/14,MSA14(1),27;  
RESOURCE/15,MSA15(1),27;  
RESOURCE/16,MSA16(1),27;  
RESOURCE/17,MSA17(1),27;  
RESOURCE/18,MSA18(1),27;  
RESOURCE/19,MSA19(1),27;  
RESOURCE/20,MSA20(1),27;  
RESOURCE/21,MSA21(1),27;  
RESOURCE/22,MSA22(1),27;  
RESOURCE/23,MSA23(1),27;  
RESOURCE/24,MSA24(1),27;  
RESOURCE/25,MSA25(1),27;  
RESOURCE/26,MSA26(1),27;  
RESOURCE/27,MSA27(1),27;  
RESOURCE/28,MSA28(1),27;  
RESOURCE/29,MSA29(1),27;  
RESOURCE/30,MSA30(1),27;  
RESOURCE/31,MSA31(1),27;  
RESOURCE/32,MSA32(1),27;  
RESOURCE/33,MSA33(1),27;  
RESOURCE/34,MSA34(1),27;  
RESOURCE/35,MSA35(1),27;  
RESOURCE/36,MSA36(1),27;  
RESOURCE/37,MSA37(1),27;

•  
•  
•  
•  
•  
•  
•  
•

⋮

RESOURCE/121,ROW12(1),18;  
 RESOURCE/122,ROW13(1),18;  
 RESOURCE/123,ROW14(1),18;  
 RESOURCE/124,ROW15(1),18;  
 RESOURCE/125,ROW16(1),18;  
 RESOURCE/126,ROW17(1),18;  
 RESOURCE/127,ROW18(1),18;  
 RESOURCE/128,ROW19(1),18;  
 RESOURCE/129,ROW20(1),18;  
 RESOURCE/130,ROW21(1),18;  
 RESOURCE/131,ROW22(1),18;  
 RESOURCE/132,ROW23(1),18;  
 RESOURCE/133,ROW24(1),18;  
 RESOURCE/134,ROW25(1),18;  
 RESOURCE/135,ROW26(1),18;  
 RESOURCE/136,ROW27(1),18;  
 RESOURCE/137,ROW28(1),18;  
 RESOURCE/138,ROW29(1),18;  
 RESOURCE/139,ROW30(1),18;  
 RESOURCE/140,ROW31(1),18;  
 RESOURCE/141,ROW32(1),18;  
 RESOURCE/142,ROW33(1),18;  
 RESOURCE/143,ROW34(1),18;  
 RESOURCE/144,ROW35(1),18;  
 RESOURCE/145,ROW36(1),18;  
 RESOURCE/146,ROW37(1),18;  
 RESOURCE/147,ROW38(1),18;  
 RESOURCE/148,ROW39(1),18;  
 RESOURCE/149,ROW40(1),18;  
 RESOURCE/150,ROW41(1),18;  
 RESOURCE/151,ROW42(1),18;  
 RESOURCE/152,ROW43(1),18;  
 RESOURCE/153,ROW44(1),18;  
 RESOURCE/154,ROW45(1),18;  
 RESOURCE/155,ROW46(1),18;  
 RESOURCE/156,ROW47(1),18;  
 RESOURCE/157,ROW48(1),18;  
 RESOURCE/158,ROW49(1),18;  
 RESOURCE/159,ROW50(1),18;  
 RESOURCE/160,ROW51(1),18;  
 RESOURCE/161,ROW52(1),18;  
 RESOURCE/162,ROW53(1),18;  
 RESOURCE/163,ROW54(1),18;  
 RESOURCE/164,ROW55(1),18;  
 RESOURCE/165,ROW56(1),18;  
 RESOURCE/166,ROW57(1),18;  
 RESOURCE/167,ROW58(1),18;  
 RESOURCE/168,ROW59(1),18;  
 RESOURCE/169,ROW60(1),18;

;  
 ;  
 ;  
 ;  
 ;

CON AND BU RESOURCES REPRESENT THE CONSOLIDATION AND  
 BUFFER STANDS IN THE CONSOLIDATION AREA. IF NNRSC = 1,  
 THE STAND IS AVAILABLE. IF NNRSC = 0, THE STAND IS  
 FILLED OR IS ABOUT TO BE FILLED.

```

RESOURCE/180,CON1(1),20;
RESOURCE/181,CON2(1),20;
RESOURCE/182,CON3(1),20;
RESOURCE/183,CON4(1),20;
RESOURCE/184,CON5(1),20;
RESOURCE/185,CON6(1),20;
RESOURCE/186,CON7(1),20;
RESOURCE/187,CON8(1),20;
RESOURCE/188,CON9(1),20;
RESOURCE/189,CON10(1),20;
RESOURCE/190,CON11(1),20;
RESOURCE/191,CON12(1),20;
RESOURCE/192,CON13(1),20;
RESOURCE/193,CON14(1),20;
RESOURCE/194,CON15(1),20;
RESOURCE/195,CON16(1),20;
RESOURCE/196,CON17(1),20;
RESOURCE/197,CON18(1),20;
RESOURCE/198,CON19(1),20;
RESOURCE/199,CON20(1),20;
RESOURCE/200,CON21(1),20;
RESOURCE/201,CON22(1),20;
RESOURCE/202,CON23(1),20;
RESOURCE/203,CON24(1),20;
RESOURCE/204,CON25(1),20;
RESOURCE/205,CON26(1),20;
RESOURCE/206,CON27(1),20;
RESOURCE/207,CON28(1),20;
RESOURCE/208,CON29(1),20;
RESOURCE/209,CON30(1),20;
RESOURCE/210,CON31(1),20;
RESOURCE/211,CON32(1),20;
RESOURCE/212,BU1(1),27;
RESOURCE/213,BU2(1),27;
RESOURCE/214,BU3(1),27;
RESOURCE/215,BU4(1),27;
RESOURCE/216,BU5(1),27;
RESOURCE/217,BU6(1),27;
RESOURCE/218,BU7(1),27;
RESOURCE/219,BU8(1),27;
RESOURCE/220,BU9(1),27;
RESOURCE/221,BU10(1),27;

```

```

RES RESOURCES REPRESENT THE RECEIVING STANDS. IF
NNRSC = 1, THE STAND IS EMPTY. IF NNRSC = 0, THE STAND
IS FILLED OR ABOUT TO BE FILLED.

```

```

RESOURCE/240,REC1(1),15;
RESOURCE/241,REC2(1),15;
RESOURCE/242,REC3(1),15;
RESOURCE/243,REC4(1),15;
RESOURCE/244,REC5(1),15;
RESOURCE/245,REC6(1),15;

```

RESOURCE/246,REC7(1),15;  
RESOURCE/247,REC8(1),15;  
RESOURCE/248,REC9(1),15;  
RESOURCE/249,REC10(1),15;  
RESOURCE/250,REC11(1),15;  
RESOURCE/251,REC12(1),15;;

DEFINE ALL CONTROL POINTS FOR THE WAREHOUSE

MODULE LOADING AREA

VCPOINT, 1;  
VCPOINT, 2;  
VCPOINT, 3 /MLS1;  
VCPOINT, 4 /MLS2;  
VCPOINT, 5;  
VCPOINT, 6 /MLS3;  
VCPOINT, 7 /MLS4;  
VCPOINT, 8;  
VCPOINT, 9 /MLS5;  
VCPOINT, 10 /MLS6;  
VCPOINT, 11;  
VCPOINT, 12 /MLS7;  
VCPOINT, 13 /MLS8;  
VCPOINT, 14;  
VCPOINT, 15 /MLS9;  
VCPOINT, 16 /MLS10;  
VCPOINT, 17;  
VCPOINT, 18 /MLS11;  
VCPOINT, 19 /MLS12;  
VCPOINT, 20;

MODULE STORAGE AREA

VCPOINT, 21;  
VCPOINT, 22;  
VCPOINT, 23 /MSA1;  
VCPOINT, 24 /MSA2;  
VCPOINT, 25;  
VCPOINT, 26 /MSA3;  
VCPOINT, 27 /MSA4;  
VCPOINT, 28 /MSA5;  
VCPOINT, 29;  
VCPOINT, 30 /MSA6;  
VCPOINT, 31 /MSA7;  
VCPOINT, 32 /MSA8;  
VCPOINT, 33 /MSA9;  
VCPOINT, 34;  
VCPOINT, 35 /MSA10;  
VCPOINT, 36 /MSA11;  
VCPOINT, 37 /MSA12;  
VCPOINT, 38;  
VCPOINT, 39 /MSA13;  
VCPOINT, 40 /MSA14;

VCPOINT, 41 /MSA15;  
VCPOINT, 42 /MSA16;  
VCPOINT, 43;  
VCPOINT, 44 /MSA17;  
VCPOINT, 45 /MSA18;  
VCPOINT, 46 /MSA19;  
VCPOINT, 47;  
VCPOINT, 48 /MSA39;  
VCPOINT, 49 /MSA40;  
VCPOINT, 50 /MSA41;  
VCPOINT, 51 /MSA42;  
VCPOINT, 52;  
VCPOINT, 53 /MSA43;  
VCPOINT, 54 /MSA44;  
VCPOINT, 55 /MSA45;  
VCPOINT, 56;  
VCPOINT, 57 /MSA46;  
VCPOINT, 58 /MSA47;  
VCPOINT, 59 /MSA48;  
VCPOINT, 60 /MSA49;  
VCPOINT, 61;  
VCPOINT, 62 /MSA50;  
VCPOINT, 63 /MSA51;  
VCPOINT, 64 /MSA52;  
VCPOINT, 65;  
VCPOINT, 66 /MSA53;  
VCPOINT, 67 /MSA54;  
VCPOINT, 68 /MSA55;  
VCPOINT, 69 /MSA56;  
VCPOINT, 70;  
VCPOINT, 71;  
VCPOINT, 72 /MSA20;  
VCPOINT, 73 /MSA21;  
VCPOINT, 74 /MSA22;  
VCPOINT, 75 /MSA23;  
VCPOINT, 76;  
VCPOINT, 77 /MSA24;  
VCPOINT, 78 /MSA25;  
VCPOINT, 79 /MSA26;  
VCPOINT, 80;  
VCPOINT, 81 /MSA27;  
VCPOINT, 82 /MSA28;  
VCPOINT, 83 /MSA29;  
VCPOINT, 84 /MSA30;  
VCPOINT, 85;  
VCPOINT, 86 /MSA31;  
VCPOINT, 87 /MSA32;  
VCPOINT, 88 /MSA33;  
VCPOINT, 89;  
VCPOINT, 90 /MSA34;  
VCPOINT, 91 /MSA35;  
VCPOINT, 92 /MSA36;  
VCPOINT, 93 /MSA37;  
VCPOINT, 94;

VCPOINT, 95 /MSA38;  
 VCPOINT, 96 /MSA57;  
 VCPOINT, 97 /MSA58;  
 VCPOINT, 98;  
 VCPOINT, 99 /MSA59;  
 VCPOINT, 100 /MSA60;  
 VCPOINT, 101 /MSA61;  
 VCPOINT, 102 /MSA62;  
 VCPOINT, 103;  
 VCPOINT, 104 /MSA63;  
 VCPOINT, 105 /MSA64;  
 VCPOINT, 106 /MSA65;  
 VCPOINT, 107;  
 VCPOINT, 108 /MSA66;  
 VCPOINT, 109 /MSA67;  
 VCPOINT, 110;  
 VCPOINT, 111;  
 VCPOINT, 112;  
 VCPOINT, 113 /MSA70;  
 VCPOINT, 114 /MSA71;  
 VCPOINT, 115 /MSA72;  
 VCPOINT, 116;  
 VCPOINT, 117 /MSA73;  
 VCPOINT, 118 /MSA74;  
 VCPOINT, 119;

#### 3P&M

VCPOINT, 120;  
 VCPOINT, 121;  
 VCPOINT, 122;  
 VCPOINT, 123;  
 VCPOINT, 124 /PM1;  
 VCPOINT, 125 /PM2;  
 VCPOINT, 126;  
 VCPOINT, 127 /PM3;  
 VCPOINT, 128 /PM4;  
 VCPOINT, 129;  
 VCPOINT, 130 /PM5;  
 VCPOINT, 131 /PM6;  
 VCPOINT, 132;  
 VCPOINT, 133 /PM7;  
 VCPOINT, 134 /PM8;  
 VCPOINT, 135;  
 VCPOINT, 136 /PM9;  
 VCPOINT, 137;  
 VCPOINT, 138;  
 VCPOINT, 139;  
 VCPOINT, 140;

#### PALLET LOADING AREA

VCPOINT, 141;  
 VCPOINT, 142;



VCPOINT, 143;  
VCPOINT, 144;  
VCPOINT, 145;  
VCPOINT, 146;  
VCPOINT, 147;  
VCPOINT, 148; /PAL1;  
VCPOINT, 149; /PAL2;  
VCPOINT, 150;  
VCPOINT, 151 /PAL3;  
VCPOINT, 152 /PAL4;  
VCPOINT, 153;  
VCPOINT, 154 /PAL5;  
VCPOINT, 155 /PAL6;  
VCPOINT, 156;  
VCPOINT, 160;  
VCPOINT, 161;  
VCPOINT, 162;  
VCPOINT, 163 /LP1;  
VCPOINT, 164 /LP2;

SOUTH END OF HIGHRISE BIN

VCPOINT, 200;  
VCPOINT, 201;  
VCPOINT, 202 /ROW4;  
VCPOINT, 203;  
VCPOINT, 204 /ROW8;  
VCPOINT, 205;  
VCPOINT, 206 /ROW12;  
VCPOINT, 207;  
VCPOINT, 208;  
VCPOINT, 209;  
VCPOINT, 210 /ROW19;  
VCPOINT, 211;  
VCPOINT, 212 /ROW23;  
VCPOINT, 213;  
VCPOINT, 214 /ROW27;  
VCPOINT, 215;  
VCPOINT, 216;  
VCPOINT, 217;

MIDDLE OF HIGHRISE BIN

VCPOINT, 218;  
VCPOINT, 219;  
VCPOINT, 220 /ROW1;  
VCPOINT, 221 /ROW2;  
VCPOINT, 222;  
VCPOINT, 223 /ROW3;  
VCPOINT, 224;  
VCPOINT, 225 /ROW5;  
VCPOINT, 226 /ROW6;  
VCPOINT, 227;  
VCPOINT, 228 /ROW7;

VCPOINT, 229;  
VCPOINT, 230 /ROW9;  
VCPOINT, 231 /ROW10;  
VCPOINT, 232;  
VCPOINT, 233 /ROW11;  
VCPOINT, 234;  
VCPOINT, 235 /ROW13;  
VCPOINT, 236 /ROW14;  
VCPOINT, 237;  
VCPOINT, 238 /ROW15;  
VCPOINT, 239;  
VCPOINT, 240;  
VCPOINT, 241;  
VCPOINT, 242 /ROW16;  
VCPOINT, 243 /ROW17;  
VCPOINT, 244;  
VCPOINT, 245 /ROW18;  
VCPOINT, 246;  
VCPOINT, 247 /ROW20;  
VCPOINT, 248 /ROW21;  
VCPOINT, 249;  
VCPOINT, 250 /ROW22;  
VCPOINT, 251;  
VCPOINT, 252 /ROW24;  
VCPOINT, 253 /ROW25;  
VCPOINT, 254;  
VCPOINT, 255 /ROW26;  
VCPOINT, 256;  
VCPOINT, 257 /ROW28;  
VCPOINT, 258 /ROW29;  
VCPOINT, 259;  
VCPOINT, 260 /ROW30;  
VCPOINT, 261;  
VCPOINT, 262 /ROW60;  
VCPOINT, 263;  
VCPOINT, 264 /ROW59;  
VCPOINT, 265 /ROW58;  
VCPOINT, 266;  
VCPOINT, 267 /ROW56;  
VCPOINT, 268 /ROW55;  
VCPOINT, 269;  
VCPOINT, 270 /ROW54;  
VCPOINT, 271;  
VCPOINT, 272 /ROW52;  
VCPOINT, 273 /ROW51;  
VCPOINT, 274;  
VCPOINT, 275 /ROW50;  
VCPOINT, 276;  
VCPOINT, 277 /ROW48;  
VCPOINT, 278 /ROW47;  
VCPOINT, 279;  
VCPOINT, 280 /ROW46;  
VCPOINT, 281;  
VCPOINT, 282;

VCPOINT, 283;  
 VCPOINT, 284 /ROW45;  
 VCPOINT, 285 /ROW44;  
 VCPOINT, 286;  
 VCPOINT, 287 /ROW43;  
 VCPOINT, 288;  
 VCPOINT, 289 /ROW41;  
 VCPOINT, 290 /ROW40;  
 VCPOINT, 291;  
 VCPOINT, 292 /ROW39;  
 VCPOINT, 293;  
 VCPOINT, 294 /ROW37;  
 VCPOINT, 295 /ROW36;  
 VCPOINT, 296;  
 VCPOINT, 297 /ROW35;  
 VCPOINT, 298;  
 VCPOINT, 299 /ROW33;  
 VCPOINT, 300 /ROW32;  
 VCPOINT, 301;  
 VCPOINT, 302 /ROW31;  
 VCPOINT, 303;  
 VCPOINT, 304;

NORTH END

VCPOINT, 305;  
 VCPOINT, 306;  
 VCPOINT, 307;  
 VCPOINT, 308 /ROW34;  
 VCPOINT, 309;  
 VCPOINT, 310 /ROW38;  
 VCPOINT, 311;  
 VCPOINT, 312 /ROW42;  
 VCPOINT, 313;  
 VCPOINT, 314;  
 VCPOINT, 315;  
 VCPOINT, 316;  
 VCPOINT, 317 /ROW49;  
 VCPOINT, 318;  
 VCPOINT, 319 /ROW53;  
 VCPOINT, 320;  
 VCPOINT, 321 /ROW57;  
 VCPOINT, 322;  
 VCPOINT, 323;  
 VCPOINT, 324;  
 VCPOINT, 325;  
 VCPOINT, 326;  
 VCPOINT, 327;  
 VCPOINT, 328;

SHIPPING/CONSOLIDATION AREA

VCPOINT, 400;  
 VCPOINT, 401;

VCPOINT, 402;  
VCPOINT, 403;

EAST SIDE SHIPPING STATIONS (SOUTH TO NORTH)

VCPOINT, 404;  
VCPOINT, 405 /SH2;  
VCPOINT, 406 /SH4;;  
VCPOINT, 407;  
VCPOINT, 408 /SH6;  
VCPOINT, 409 /SH8;  
VCPOINT, 410;  
VCPOINT, 411 /SH10;  
VCPOINT, 412 /SH12;  
VCPOINT, 413;  
VCPOINT, 414 /SH14;  
VCPOINT, 415 /SH16;  
VCPOINT, 416;  
VCPOINT, 417 /SH18;  
VCPOINT, 418 /SH20;  
VCPOINT, 419;  
VCPOINT, 420 /SH22;  
VCPOINT, 421 /SH24;  
VCPOINT, 422;  
VCPOINT, 423 /SH26;  
VCPOINT, 424 /SH28;  
VCPOINT, 425;  
VCPOINT, 426 /SH30;  
VCPOINT, 427 /SH32;  
VCPOINT, 428;  
VCPOINT, 429;

BUFFERS

VCPOINT, 430 /BU1;  
VCPOINT, 431 /BU2;  
VCPOINT, 432;  
VCPOINT, 433 /BU3;  
VCPOINT, 434 /BU4;  
VCPOINT, 435;  
VCPOINT, 436 /BU5;  
VCPOINT, 437 /BU6;  
VCPOINT, 438;  
VCPOINT, 439 /BU7;  
VCPOINT, 440 /BU8;  
VCPOINT, 441;  
VCPOINT, 442 /BU9;  
VCPOINT, 443 /BU10;  
VCPOINT, 444;  
VCPOINT, 445;  
VCPOINT, 446;

WEST SIDE SHIPPING STATIONS (SOUTH TO NORTH)

```

VCPOINT, 447 /SH1;
VCPOINT, 448 /SH3;
VCPOINT, 449;
VCPOINT, 450 /SH5;
VCPOINT, 451 /SH7;
VCPOINT, 452;
VCPOINT, 453 /SH9;
VCPOINT, 454 /SH11;
VCPOINT, 455;
VCPOINT, 456 /SH13;
VCPOINT, 457 /SH15;
VCPOINT, 458;
VCPOINT, 459 /SH17;
VCPOINT, 460 /SH19;
VCPOINT, 461;
VCPOINT, 462 /SH21;
VCPOINT, 463 /SH23;
VCPOINT, 464;
VCPOINT, 465 /SH25;
VCPOINT, 466 /SH27;
VCPOINT, 467;
VCPOINT, 468 /SH29;
VCPOINT, 469 /SH31;
VCPOINT, 470;
VCPOINT, 471;
VCPOINT, 472;
VCPOINT, 473;
VCPOINT, 474;
VCPOINT, 475;
VCPOINT, 476;
VCPOINT, 477;
VCPOINT, 478;
VCPOINT, 479;
VCPOINT, 480 /PSH1;
VCPOINT, 481;
VCPOINT, 490;
VCPOINT, 491;
VCPOINT, 492;
VCPOINT, 493 /DOCK;
VCPOINT, 494;
VCPOINT, 495;
VCPOINT, 496 /LP4;
VCPOINT, 497 /LP3;
VCPOINT, 498;

```

# PALLET STORAGE CONTROL POINTS

```

VCPOINT, 500;
VCPOINT, 501 /TURN1A;
VCPOINT, 502;
VCPOINT, 503 /TURN1B;
VCPOINT, 504;
VCPOINT, 505;

```

VCPOINT, 506;  
VCPOINT, 507 /PALS14;  
VCPOINT, 508;  
VCPOINT, 509 /PALS15;  
VCPOINT, 510;  
VCPOINT, 511 /PALS16;  
VCPOINT, 512;  
VCPOINT, 513 /PALS17;  
VCPOINT, 514;  
VCPOINT, 515 /PALS18;  
VCPOINT, 516;  
VCPOINT, 517 /PALS19;  
VCPOINT, 518;  
VCPOINT, 519 /PALS20;  
VCPOINT, 520;  
VCPOINT, 521 /PALS21;  
VCPOINT, 522;  
VCPOINT, 523 /PALS22;  
VCPOINT, 524;  
VCPOINT, 525 /PALS23;  
VCPOINT, 526;  
VCPOINT, 527 /PALS24;  
VCPOINT, 528;  
VCPOINT, 529 /PALS25;  
VCPOINT, 530;  
VCPOINT, 531 /PALS26;  
VCPOINT, 532;  
VCPOINT, 533 /PALS27;  
VCPOINT, 534;  
VCPOINT, 535 /PALS28;  
VCPOINT, 536;  
VCPOINT, 537 /PALS29;  
VCPOINT, 541;  
VCPOINT, 542;  
VCPOINT, 543;  
VCPOINT, 544;  
VCPOINT, 545;  
VCPOINT, 546;  
VCPOINT, 547 /PALS1;  
VCPOINT, 548;  
VCPOINT, 549 /PALS2;  
VCPOINT, 550;  
VCPOINT, 551 /PALS3;  
VCPOINT, 552;  
VCPOINT, 553 /PALS4;  
VCPOINT, 554;  
VCPOINT, 555 /PALS5;  
VCPOINT, 556;  
VCPOINT, 557 /PALS6;  
VCPOINT, 558;  
VCPOINT, 559 /PALS7;  
VCPOINT, 560;  
VCPOINT, 561 /PALS8;  
VCPOINT, 562;

```

VCPOINT, 563  /PALS9;
VCPOINT, 564;
VCPOINT, 565  /PALS10;
VCPOINT, 566;
VCPOINT, 567  /PALS11;
VCPOINT, 568;
VCPOINT, 569  /PALS12;
VCPOINT, 570;
VCPOINT, 571  /PALS13;
VCPOINT, 575;
VCPOINT, 576;
VCPOINT, 577;
VCPOINT, 578;
VCPOINT, 579  /PBUF1;
VCPOINT, 580  /PBUF2;
VCPOINT, 581;
VCPOINT, 582  /PBUF3;
VCPOINT, 583  /PBUF4;
VCPOINT, 586;
VCPOINT, 587;
VCPOINT, 588  /TURN2A;
VCPOINT, 589;
VCPOINT, 590  /TURN2B;

```

```

VSEGMENT, SEGMENT */LABEL, BEGINNING CONTROL POINT, ENDING
CONTROL POINT, DISTANCE, DIRECTION OF SEGMENT

```

#### SEGMENTS IN RECEIVING

```

VSGMENT, 01,      1, 20, 109, UNI;
VSGMENT, 02,      1,  2,  25, UNI;
VSGMENT, 03,      2,  5,  12, UNI;
VSGMENT, 04,      5,  8,  12, UNI;
VSGMENT, 05,      8, 11,  14, UNI;
VSGMENT, 06,     11, 14,  12, UNI;
VSGMENT, 07,     14, 17,  12, UNI;
VSGMENT, 08/MLS1,  2,  3,  10, B1, 1;
VSGMENT, 09/MLS2,  2,  4,  10, B1, 1;
VSGMENT, 10/MLS3,  5,  6,  10, B1, 1;
VSGMENT, 11/MLS4,  5,  7,  10, B1, 1;
VSGMENT, 12/MLS5,  8,  9,  10, B1, 1;
VSGMENT, 13/MLS6,  8, 10,  10, B1, 1;
VSGMENT, 14/MLS7, 11, 12,  10, B1, 1;
VSGMENT, 15/MLS8, 11, 13,  10, B1, 1;
VSGMENT, 16/MLS9, 14, 15,  10, B1, 1;
VSGMENT, 17/MLS10, 14, 16,  10, B1, 1;
VSGMENT, 18/MLS11, 17, 18,  10, B1, 1;
VSGMENT, 19/MLS12, 17, 19,  10, B1, 1;
VSGMENT, 20,     17, 20,  31, UNI;

```

#### MODULE STORAGE AREA

```

VSGMENT, 21,      20, 21,  10, UNI;
VSGMENT, 22,      21, 22,  10, UNI;

```

VSGMENT,	23,	22,	25,	10,	UNI;
VSGMENT,	24,	25,	29,	10,	UNI;
VSGMENT,	25,	29,	34,	10,	UNI;
VSGMENT,	26,	34,	38,	10,	UNI;
VSGMENT,	27,	38,	43,	10,	UNI;
VSGMENT,	28,	43,	47,	11,	UNI;
VSGMENT,	29,	47,	52,	10,	UNI;
VSGMENT,	30,	52,	56,	10,	UNI;
VSGMENT,	31,	56,	61,	10,	UNI;
VSGMENT,	32,	61,	65,	10,	UNI;
VSGMENT,	33/MSA1,	22,	23,	10,	BI, 1;
VSGMENT,	34/MSA2,	22,	24,	10,	BI, 1;
VSGMENT,	35/MSA3,	25,	26,	10,	BI, 1;
VSGMENT,	36/MSA4,	25,	27,	10,	BI, 1;
VSGMENT,	37/MSA5,	25,	28,	10,	BI, 1;
VSGMENT,	38/MSA6,	29,	30,	10,	BI, 1;
VSGMENT,	39/MSA7,	29,	31,	10,	BI, 1;
VSGMENT,	40/MSA8,	29,	32,	10,	BI, 1;
VSGMENT,	41/MSA9,	29,	33,	10,	BI, 1;
VSGMENT,	42/MSA10,	34,	35,	10,	BI, 1;
VSGMENT,	43/MSA11,	34,	36,	10,	BI, 1;
VSGMENT,	44/MSA12,	34,	37,	10,	BI, 1;
VSGMENT,	45/MSA13,	38,	39,	10,	BI, 1;
VSGMENT,	46/MSA14,	38,	40,	10,	BI, 1;
VSGMENT,	47/MSA15,	38,	41,	10,	BI, 1;
VSGMENT,	48/MSA16,	38,	42,	10,	BI, 1;
VSGMENT,	49/MSA17,	43,	44,	10,	BI, 1;
VSGMENT,	50/MSA18,	43,	45,	10,	BI, 1;
VSGMENT,	51/MSA19,	43,	46,	10,	BI, 1;
VSGMENT,	52/MSA39,	47,	48,	10,	BI, 1;
VSGMENT,	53/MSA40,	47,	49,	10,	BI, 1;
VSGMENT,	54/MSA41,	47,	50,	10,	BI, 1;
VSGMENT,	55/MSA42,	47,	51,	10,	BI, 1;
VSGMENT,	56/MSA43,	52,	53,	10,	BI, 1;
VSGMENT,	57/MSA44,	52,	54,	10,	BI, 1;
VSGMENT,	58/MSA45,	52,	55,	10,	BI, 1;
VSGMENT,	59/MSA46,	56,	57,	10,	BI, 1;
VSGMENT,	60/MSA47,	56,	58,	10,	BI, 1;
VSGMENT,	61/MSA48,	56,	59,	10,	BI, 1;
VSGMENT,	62/MSA49,	56,	60,	10,	BI, 1;
VSGMENT,	63/MSA50,	61,	62,	10,	BI, 1;
VSGMENT,	64/MSA51,	61,	63,	10,	BI, 1;
VSGMENT,	65/MSA52,	61,	64,	10,	BI, 1;
VSGMENT,	66/MSA53,	65,	66,	10,	BI, 1;
VSGMENT,	67/MSA54,	65,	67,	10,	BI, 1;
VSGMENT,	68/MSA55,	65,	68,	10,	BI, 1;
VSGMENT,	69/MSA56,	65,	69,	10,	BI, 1;
VSGMENT,	70,	65,	70,	10,	UNI;
VSGMENT,	71,	70,	119,	10,	UNI;
VSGMENT,	72,	22,	71,	27,	UNI;
VSGMENT,	73,	71,	76,	10,	UNI;
VSGMENT,	74,	76,	80,	10,	UNI;
VSGMENT,	75,	80,	85,	10,	UNI;
VSGMENT,	76,	85,	89,	10,	UNI;



VSGMENT, 77,	89,	94,	11,	UN1;
VSGMENT, 78,	94,	98,	10,	UN1;
VSGMENT, 79,	98,	103,	10,	UN1;
VSGMENT, 80,	103,	107,	10,	UN1;
VSGMENT, 81,	107,	112,	10,	UN1;
VSGMENT, 82,	112,	116,	10,	UN1;
VSGMENT, 83/MSA20,	71,	72,	10,	B1, 1;
VSGMENT, 84/MSA21,	71,	73,	10,	B1, 1;
VSGMENT, 85/MSA22,	71,	74,	10,	B1, 1;
VSGMENT, 86/MSA23,	71,	75,	10,	B1, 1;
VSGMENT, 87/MSA24,	76,	77,	10,	B1, 1;
VSGMENT, 88/MSA25,	76,	78,	10,	B1, 1;
VSGMENT, 89/MSA26,	76,	79,	10,	B1, 1;
VSGMENT, 90/MSA27,	80,	81,	10,	B1, 1;
VSGMENT, 91/MSA28,	80,	82,	10,	B1, 1;
VSGMENT, 92/MSA29,	80,	83,	10,	B1, 1;
VSGMENT, 93/MSA30,	80,	84,	10,	B1, 1;
VSGMENT, 94/MSA31,	85,	86,	10,	B1, 1;
VSGMENT, 95/MSA32,	85,	87,	10,	B1, 1;
VSGMENT, 96/MSA33,	85,	88,	10,	B1, 1;
VSGMENT, 97/MSA34,	89,	90,	10,	B1, 1;
VSGMENT, 98/MSA35,	89,	91,	10,	B1, 1;
VSGMENT, 99/MSA36,	89,	92,	10,	B1, 1;
VSGMENT, 100/MSA37,	89,	93,	10,	B1, 1;
VSGMENT, 101/MSA38,	94,	95,	10,	B1, 1;
VSGMENT, 102/MSA57,	94,	96,	10,	B1, 1;
VSGMENT, 103/MSA58,	94,	97,	10,	B1, 1;
VSGMENT, 104/MSA59,	98,	99,	10,	B1, 1;
VSGMENT, 105/MSA60,	98,	100,	10,	B1, 1;
VSGMENT, 106/MSA61,	98,	101,	10,	B1, 1;
VSGMENT, 107/MSA62,	98,	102,	10,	B1, 1;
VSGMENT, 108/MSA63,	103,	104,	10,	B1, 1;
VSGMENT, 109/MSA64,	103,	105,	10,	B1, 1;
VSGMENT, 110/MSA65,	103,	106,	10,	B1, 1;
VSGMENT, 111/MSA66,	107,	108,	10,	B1, 1;
VSGMENT, 112/MSA67,	107,	109,	10,	B1, 1;
VSGMENT, 113/MSA68,	107,	110,	10,	B1, 1;
VSGMENT, 114/MSA69,	107,	111,	10,	B1, 1;
VSGMENT, 115/MSA70,	112,	113,	10,	B1, 1;
VSGMENT, 116/MSA71,	112,	114,	10,	B1, 1;
VSGMENT, 117/MSA72,	112,	115,	10,	B1, 1;
VSGMENT, 118/MSA73,	116,	117,	10,	B1, 1;
VSGMENT, 119/MSA74,	116,	118,	10,	B1, 1;
VSGMENT, 120,	116,	70,	31,	UN1;

3P&M

VSGMENT, 121,	20,	120,	10,	UN1;
VSGMENT, 122,	120,	139,	10,	UN1;
VSGMENT, 123,	120,	160,	52,	UN1;
VSGMENT, 124,	121,	122,	137,	UN1;
VSGMENT, 125,	122,	123,	72,	UN1;
VSGMENT, 126,	123,	126,	12,	UN1;
VSGMENT, 127,	126,	129,	15,	UN1;

VSGMENT, 128,	129, 132,	15, UNI;
VSGMENT, 129,	132, 135,	12, UNI;
VSGMENT, 130/PM1,	123, 124,	10, B1, 1;
VSGMENT, 131/PM2,	123, 125,	10, B1, 1;
VSGMENT, 132/PM3,	126, 127,	10, B1, 1;
VSGMENT, 133/PM4,	126, 128,	10, B1, 1;
VSGMENT, 134/PM5,	129, 130,	10, B1, 1;
VSGMENT, 135/PM6,	129, 131,	10, B1, 1;
VSGMENT, 136/PM7,	132, 133,	10, B1, 1;
VSGMENT, 137/PM8,	132, 134,	10, B1, 1;
VSGMENT, 138/PM9,	135, 136,	10, B1, 1;
VSGMENT, 139,	135, 137,	26, UNI;
VSGMENT, 140,	137, 122,	10, UNI;
VSGMENT, 141,	137, 138, 135,	UNI;
VSGMENT, 142,	138, 162,	35, UNI;
VSGMENT, 143,	138, 161,	10, UNI;
VSGMENT, 144,	139, 21,	13, UNI;
VSGMENT, 145,	160, 121,	10, UNI;
VSGMENT, 146,	160, 161,	10, UNI;
VSGMENT, 147,	161, 139,	54, UNI;
VSGMENT, 148,	162, 140,	10, UNI;
VSGMENT, 149/LP1,	161, 163,	48, UNI;
VSGMENT, 150,	163, 162,	10, UNI;

#### PALLET LOADING

VSGMENT, 151,	119, 141,	82, UNI;
VSGMENT, 152,	141, 142,	10, UNI;
VSGMENT, 153,	142, 143,	10, UNI;
VSGMENT, 154,	143, 144,	10, UNI;
VSGMENT, 155,	144, 156,	10, UNI;
VSGMENT, 156,	156, 1,	88, UNI;
VSGMENT, 157,	145, 146,	12, UNI;
VSGMENT, 158/LP2,	146, 164,	110, UNI;
VSGMENT, 159,	146, 147,	79, UNI;
VSGMENT, 160,	147, 150,	11, UNI;
VSGMENT, 161,	150, 153,	15, UNI;
VSGMENT, 162/PAL1,	147, 148,	11, B1, 1;
VSGMENT, 163/PAL2,	147, 149,	11, B1, 1;
VSGMENT, 164/PAL3,	150, 151,	11, B1, 1;
VSGMENT, 165/PAL4,	150, 152,	11, B1, 1;
VSGMENT, 166/PAL5,	153, 154,	11, B1, 1;
VSGMENT, 167/PAL6,	153, 155,	11, B1, 1;
VSGMENT, 168,	153, 156,	16, UNI;
VSGMENT, 169,	156, 141,	10, UNI;
VSGMENT, 170,	144, 142,	10, UNI;
VSGMENT, 171,	164, 144,	10, UNI;
VSGMENT, 172,	119, 1,	10, UNI;
VSGMENT, 173,	21, 119, 113,	UNI;

#### HIGHRISE BIN STORAGE AREA

VSGMENT, 200,	140, 200, 197,	UNI;
VSGMENT, 201,	200, 201, 23,	UNI;

VSGMENT, 202,	201, 203,	34, UNI;
VSGMENT, 203,	203, 205,	34, UNI;
VSGMENT, 204/ROW4,	201, 202,	14, B1, 1;
VSGMENT, 205/ROW8,	203, 204,	14, B1, 1;
VSGMENT, 206/ROW12,	205, 206,	14, B1, 1;
VSGMENT, 207,	205, 207,	41, UNI;
VSGMENT, 208,	207, 217,	10, UNI;
VSGMENT, 209,	207, 208,	15, UNI;
VSGMENT, 210,	208, 209,	23, UNI;
VSGMENT, 211,	209, 211,	34, UNI;
VSGMENT, 212,	211, 213,	34, UNI;
VSGMENT, 213/ROW19,	209, 210,	14, B1, 1;
VSGMENT, 214/ROW23,	211, 212,	14, B1, 1;
VSGMENT, 215/ROW27,	213, 214,	14, B1, 1;
VSGMENT, 216,	213, 215,	39, UNI;
VSGMENT, 217,	215, 216,	133, UNI;
VSGMENT, 218,	216, 208,	10, UNI;
VSGMENT, 219,	216, 217,	15, UNI;
VSGMENT, 220,	217, 200,	133, UNI;

# MIDDLE SECTION

VSGMENT, 221,	140, 218,	69, UNI;
VSGMENT, 222,	218, 303,	19, UNI;
VSGMENT, 223,	218, 219,	17, UNI;
VSGMENT, 224,	219, 222,	16, UNI;
VSGMENT, 225,	222, 224,	17, UNI;
VSGMENT, 226,	224, 227,	16, UNI;
VSGMENT, 227,	227, 229,	17, UNI;
VSGMENT, 349,	229, 232,	16, UNI;
VSGMENT, 228,	232, 234,	17, UNI;
VSGMENT, 229,	234, 237,	13, UNI;
VSGMENT, 230/ROW1,	219, 220,	17, B1, 1;
VSGMENT, 231/ROW2,	219, 221,	17, B1, 1;
VSGMENT, 232/ROW3,	222, 223,	17, B1, 1;
VSGMENT, 233/ROW5,	224, 225,	17, B1, 1;
VSGMENT, 234/ROW6,	224, 226,	17, B1, 1;
VSGMENT, 235/ROW7,	227, 228,	17, B1, 1;
VSGMENT, 236/ROW9,	229, 230,	17, B1, 1;
VSGMENT, 237/ROW10,	229, 231,	17, B1, 1;
VSGMENT, 238/ROW11,	232, 233,	17, B1, 1;
VSGMENT, 239/ROW13,	234, 235,	17, B1, 1;
VSGMENT, 240/ROW14,	234, 236,	17, B1, 1;
VSGMENT, 241/ROW15,	237, 238,	17, B1, 1;
VSGMENT, 242,	237, 283,	12, UNI;
VSGMENT, 243,	237, 239,	10, UNI;
VSGMENT, 244,	239, 207,	266, UNI;
VSGMENT, 245,	239, 240,	12, UNI;
VSGMENT, 246,	240, 281,	12, UNI;
VSGMENT, 247,	240, 241,	10, UNI;
VSGMENT, 248,	241, 244,	16, UNI;
VSGMENT, 249,	244, 246,	17, UNI;
VSGMENT, 250,	246, 249,	16, UNI;
VSGMENT, 251,	249, 251,	17, UNI;

VSGMENT, 252,	251, 254,	16, UN1;
VSGMENT, 253,	254, 256,	17, UN1;
VSGMENT, 254,	256, 263,	26, UN1;
VSGMENT, 255,	256, 259,	12, UN1;
VSGMENT, 256/ROW16,	241, 242,	17, B1, 1;
VSGMENT, 257/ROW17,	241, 243,	17, B1, 1;
VSGMENT, 258/ROW18,	244, 245,	17, B1, 1;
VSGMENT, 259/ROW20,	246, 247,	17, B1, 1;
VSGMENT, 260/ROW21,	246, 248,	17, B1, 1;
VSGMENT, 261/ROW22,	249, 250,	17, B1, 1;
VSGMENT, 262/ROW24,	251, 252,	17, B1, 1;
VSGMENT, 263/ROW25,	251, 253,	17, B1, 1;
VSGMENT, 264/ROW26,	254, 255,	17, B1, 1;
VSGMENT, 265/ROW28,	256, 257,	17, B1, 1;
VSGMENT, 266/ROW29,	256, 258,	17, B1, 1;
VSGMENT, 267/ROW30,	259, 260,	17, B1, 1;
VSGMENT, 268,	259, 304,	10, UN1;
VSGMENT, 269,	215, 304, 270,	UN1;
VSGMENT, 270,	304, 261,	10, UN1;
VSGMENT, 271,	261, 263,	10, UN1;
VSGMENT, 272,	263, 266,	23, UN1;
VSGMENT, 273,	266, 269,	16, UN1;
VSGMENT, 274,	269, 271,	17, UN1;
VSGMENT, 275,	271, 274,	16, UN1;
VSGMENT, 276,	274, 276,	17, UN1;
VSGMENT, 277,	276, 279,	16, UN1;
VSGMENT, 278/ROW60,	261, 262,	17, B1, 1;
VSGMENT, 279/ROW59,	263, 264,	17, B1, 1;
VSGMENT, 280/ROW58,	263, 265,	17, B1, 1;
VSGMENT, 281/ROW56,	266, 267,	17, B1, 1;
VSGMENT, 282/ROW55,	266, 268,	17, B1, 1;
VSGMENT, 283/ROW54,	269, 270,	17, B1, 1;
VSGMENT, 284/ROW52,	271, 272,	17, B1, 1;
VSGMENT, 285/ROW51,	271, 273,	17, B1, 1;
VSGMENT, 286/ROW50,	274, 275,	17, B1, 1;
VSGMENT, 287/ROW48,	276, 277,	17, B1, 1;
VSGMENT, 288/ROW47,	276, 278,	17, B1, 1;
VSGMENT, 289/ROW46,	279, 280,	17, B1, 1;
VSGMENT, 290,	279, 281,	10, UN1;
VSGMENT, 291,	281, 282,	12, UN1;
VSGMENT, 292,	282, 239,	12, UN1;
VSGMENT, 293,	282, 283,	10, UN1;
VSGMENT, 294,	283, 286,	10, UN1;
VSGMENT, 295,	286, 288,	17, UN1;
VSGMENT, 296,	288, 291,	16, UN1;
VSGMENT, 297,	291, 293,	17, UN1;
VSGMENT, 298,	293, 296,	16, UN1;
VSGMENT, 299,	296, 298,	17, UN1;
VSGMENT, 300,	298, 301,	16, UN1;
VSGMENT, 301/ROW45,	283, 284,	17, B1, 1;
VSGMENT, 302/ROW44,	283, 285,	17, B1, 1;
VSGMENT, 303/ROW43,	286, 287,	17, B1, 1;
VSGMENT, 304/ROW41,	288, 289,	17, B1, 1;
VSGMENT, 305/ROW40,	288, 290,	17, B1, 1;

VSGMENT, 306/ROW39,	291, 292,	17,	BI,	1;
VSGMENT, 307/ROW37,	293, 294,	17,	BI,	1;
VSGMENT, 308/ROW36,	293, 295,	17,	BI,	1;
VSGMENT, 309/ROW35,	296, 297,	17,	BI,	1;
VSGMENT, 310/ROW33,	298, 299,	17,	BI,	1;
VSGMENT, 311/ROW32,	298, 300,	17,	BI,	1;
VSGMENT, 312/ROW31,	301, 302,	17,	BI,	1;
VSGMENT, 313,	301, 219,	12,	UN1;	
VSGMENT, 314,	301, 303,	19,	UN1;	
VSGMENT, 315,	303, 143, 149,		UN1;	

#### NORTH END OF HIGHRISE

VSGMENT, 316,	143, 305, 103,		UN1;	
VSGMENT, 317,	305, 145,	10,	UN1;	
VSGMENT, 318,	305, 306,	12,	UN1;	
VSGMENT, 319,	306, 307,	23,	UN1;	
VSGMENT, 320,	307, 309,	34,	UN1;	
VSGMENT, 321,	309, 311,	34,	UN1;	
VSGMENT, 322/ROW34,	307, 308,	14,	BI,	1;
VSGMENT, 323/ROW38,	309, 310,	14,	BI,	1;
VSGMENT, 324/ROW42,	311, 312,	14,	BI,	1;
VSGMENT, 325,	311, 313,	35,	UN1;	
VSGMENT, 326,	313, 314,	10,	UN1;	
VSGMENT, 327,	314, 326,	10,	UN1;	
VSGMENT, 328,	314, 282, 265,		UN1;	
VSGMENT, 329,	314, 315,	11,	UN1;	
VSGMENT, 330,	315, 316,	23,	UN1;	
VSGMENT, 331,	316, 318,	34,	UN1;	
VSGMENT, 332,	318, 320,	34,	UN1;	
VSGMENT, 333/ROW49,	316, 317,	14,	BI,	1;
VSGMENT, 334/ROW53,	318, 319,	14,	BI,	1;
VSGMENT, 335/ROW57,	320, 321,	14,	BI,	1;
VSGMENT, 336,	320, 322,	34,	UN1;	
VSGMENT, 337,	304, 322, 279,		UN1;	
VSGMENT, 338,	322, 323, 130,		UN1;	
VSGMENT, 339,	323, 315,	10,	UN1;	
VSGMENT, 340,	323, 324,	11,	UN1;	
VSGMENT, 341,	324, 325,	10,	UN1;	
VSGMENT, 342,	324, 326,	10,	UN1;	
VSGMENT, 343,	326, 325,	10,	UN1;	
VSGMENT, 344,	326, 327,	10,	UN1;	
VSGMENT, 345,	327, 313,	10,	UN1;	
VSGMENT, 346,	327, 328, 121,		UN1;	
VSGMENT, 347,	328, 306,	10,	UN1;	
VSGMENT, 348,	328, 145,	16,	UN1;	

#### SHIPPING WAREHOUSE

VSGMENT, 400,	325, 491, 187,		UN1;	
VSGMENT, 401,	400, 445,	10,	UN1;	
VSGMENT, 402,	400, 401,	40,	UN1;	
VSGMENT, 403,	401, 402,	55,	UN1;	

VSGMENT, 404,	402, 498,	10, UNI;
VSGMENT, 405,	402, 403,	54, UNI;
VSGMENT, 406,	403, 490,	10, UNI;
VSGMENT, 407,	404, 407,	12, UNI;
VSGMENT, 408,	407, 410,	12, UNI;
VSGMENT, 409,	410, 413,	12, UNI;
VSGMENT, 410,	413, 416,	12, UNI;
VSGMENT, 411,	416, 419,	12, UNI;
VSGMENT, 412,	419, 422,	12, UNI;
VSGMENT, 413,	422, 425,	12, UNI;
VSGMENT, 414/SH2,	404, 405,	10, BI, 1;
VSGMENT, 415/SH4,	404, 406,	10, BI, 1;
VSGMENT, 416/SH6,	407, 408,	10, BI, 1;
VSGMENT, 417/SH86,	407, 409,	10, BI, 1;
VSGMENT, 418/SH10,	410, 411,	10, BI, 1;
VSGMENT, 419/SH12,	410, 412,	10, BI, 1;
VSGMENT, 420/SH14,	413, 414,	10, BI, 1;
VSGMENT, 421/SH16,	413, 415,	10, BI, 1;
VSGMENT, 422/SH18,	416, 417,	10, BI, 1;
VSGMENT, 423/SH20,	416, 418,	10, BI, 1;
VSGMENT, 424/SH22,	419, 420,	10, BI, 1;
VSGMENT, 425/SH24,	419, 421,	10, BI, 1;
VSGMENT, 426/SH26,	422, 423,	10, BI, 1;
VSGMENT, 427/SH28,	422, 424,	10, BI, 1;
VSGMENT, 428/SH30,	425, 426,	10, BI, 1;
VSGMENT, 429/SH32,	425, 427,	10, BI, 1;
VSGMENT, 430,	425, 428,	24, UNI;
VSGMENT, 431,	428, 403,	10, UNI;
VSGMENT, 432,	428, 498,	46, UNI;
VSGMENT, 433,	429, 432,	10, UNI;
VSGMENT, 434,	432, 435,	10, UNI;
VSGMENT, 435,	435, 438,	10, UNI;
VSGMENT, 436,	438, 441,	10, UNI;
VSGMENT, 437/BU1,	429, 430,	10, BI, 1;
VSGMENT, 438/BU2,	429, 431,	10, BI, 1;
VSGMENT, 439/BU3,	432, 433,	10, BI, 1;
VSGMENT, 440/BU4,	432, 434,	10, BI, 1;
VSGMENT, 441/BU5,	435, 436,	10, BI, 1;
VSGMENT, 442/BU6,	435, 437,	10, BI, 1;
VSGMENT, 443/BU7,	438, 439,	10, BI, 1;
VSGMENT, 444/BU8,	438, 440,	10, BI, 1;
VSGMENT, 445/BU9,	441, 442,	10, BI, 1;
VSGMENT, 446/BU10,	441, 443,	10, BI, 1;
VSGMENT, 447,	441, 444,	12, UNI;
VSGMENT, 448,	444, 401,	10, UNI;
VSGMENT, 449,	444, 445,	40, UNI;
VSGMENT, 450,	445, 478,	10, UNI;
VSGMENT, 451,	445, 446,	18, UNI;
VSGMENT, 452,	446, 449,	16, UNI;
VSGMENT, 453,	449, 452,	12, UNI;
VSGMENT, 454,	452, 455,	12, UNI;
VSGMENT, 455,	455, 458,	16, UNI;
VSGMENT, 456,	458, 461,	12, UNI;
VSGMENT, 457,	461, 464,	12, UNI;



VSGMENT, 553,	504, 505, 171, UN1;
VSGMENT, 554,	505, 506, 36, UN1;
VSGMENT, 555,	506, 508, 11, UN1;
VSGMENT, 556,	508, 510, 11, UN1;
VSGMENT, 557,	510, 512, 11, UN1;
VSGMENT, 558,	512, 514, 11, UN1;
VSGMENT, 559,	514, 516, 11, UN1;
VSGMENT, 560,	516, 518, 13, UN1;
VSGMENT, 561,	518, 520, 13, UN1;
VSGMENT, 562,	520, 522, 13, UN1;
VSGMENT, 563,	522, 524, 13, UN1;
VSGMENT, 564,	524, 526, 13, UN1;
VSGMENT, 565,	526, 528, 13, UN1;
VSGMENT, 566,	528, 530, 13, UN1;
VSGMENT, 567,	530, 532, 13, UN1;
VSGMENT, 568,	532, 534, 13, UN1;
VSGMENT, 569,	534, 536, 13, UN1;
VSGMENT, 570/PALS14,	506, 507, 10, B1, 1;
VSGMENT, 571/PALS15,	508, 509, 10, B1, 1;
VSGMENT, 572/PALS16,	510, 511, 10, B1, 1;
VSGMENT, 573/PALS17,	512, 513, 10, B1, 1;
VSGMENT, 574/PALS18,	514, 515, 10, B1, 1;
VSGMENT, 575/PALS19,	516, 517, 10, B1, 1;
VSGMENT, 576/PALS20,	518, 519, 10, B1, 1;
VSGMENT, 577/PALS21,	520, 521, 10, B1, 1;
VSGMENT, 578/PALS22,	522, 523, 10, B1, 1;
VSGMENT, 579/PALS23,	524, 525, 10, B1, 1;
VSGMENT, 580/PALS24,	526, 527, 10, B1, 1;
VSGMENT, 581/PALS25,	528, 529, 10, B1, 1;
VSGMENT, 582/PALS26,	530, 531, 10, B1, 1;
VSGMENT, 583/PALS27,	532, 533, 10, B1, 1;
VSGMENT, 584/PALS28,	534, 535, 10, B1, 1;
VSGMENT, 585/PALS29,	536, 537, 10, B1, 1;
VSGMENT, 586,	510, 543, 10, UN1;
VSGMENT, 587,	542, 516, 10, UN1;
VSGMENT, 588,	526, 541, 10, UN1;
VSGMENT, 589,	536, 541, 73, UN1;
VSGMENT, 590,	541, 542, 64, UN1;
VSGMENT, 591,	542, 543, 29, UN1;
VSGMENT, 592,	543, 505, 10, UN1;
VSGMENT, 593,	543, 544, 175, UN1;
VSGMENT, 594,	544, 545, 10, UN1;
VSGMENT, 595,	545, 586, 10, UN1;
VSGMENT, 596,	545, 546, 20, UN1;
VSGMENT, 597,	546, 548, 10, UN1;
VSGMENT, 598,	548, 550, 13, UN1;
VSGMENT, 599,	550, 552, 13, UN1;
VSGMENT, 600,	552, 554, 13, UN1;
VSGMENT, 601,	554, 556, 13, UN1;
VSGMENT, 602,	556, 558, 13, UN1;
VSGMENT, 603,	558, 560, 13, UN1;
VSGMENT, 604,	560, 562, 13, UN1;
VSGMENT, 605,	562, 564, 13, UN1;
VSGMENT, 606,	564, 566, 13, UN1;



```

VSGMENT, 607,      566, 568, 13, UNI;
VSGMENT, 608,      568, 570, 13, UNI;
VSGMENT, 609/PALS1, 546, 547, 10, B1, 1;
VSGMENT, 610/PALS2, 548, 549, 10, B1, 1;
VSGMENT, 611/PALS3, 550, 551, 10, B1, 1;
VSGMENT, 612/PALS4, 552, 553, 10, B1, 1;
VSGMENT, 613/PALS5, 554, 555, 10, B1, 1;
VSGMENT, 614/PALS6, 556, 557, 10, B1, 1;a
VSGMENT, 615/PALS7, 558, 559, 10, B1, 1;
VSGMENT, 616/PALS8, 560, 561, 10, B1, 1;
VSGMENT, 617/PALS9, 562, 563, 10, B1, 1;
VSGMENT, 618/PALS10, 564, 565, 10, B1, 1;
VSGMENT, 619/PALS11, 566, 567, 10, B1, 1;
VSGMENT, 620/PALS12, 568, 569, 10, B1, 1;
VSGMENT, 621/PALS13, 570, 571, 10, B1, 1;
VSGMENT, 622,      548, 577, 10, UNI;
VSGMENT, 623,      576, 552, 10, UNI;
VSGMENT, 624,      560, 575, 10, UNI;
VSGMENT, 625,      570, 575, 75, UNI;
VSGMENT, 626,      575, 576, 65, UNI;
VSGMENT, 627,      576, 577, 16, UNI;
VSGMENT, 628,      577, 578, 20, UNI;
VSGMENT, 629,      578, 581, 10, UNI;
VSGMENT, 630/PBUF1, 578, 579, 10, B1, 1;
VSGMENT, 631/PBUF2, 578, 580, 10, B1, 1;
VSGMENT, 632/PBUF3, 581, 582, 10, B1, 1;
VSGMENT, 633/PBUF4, 581, 583, 10, B1, 1;
VSGMENT, 634,      581, 586, 12, UNI;
VSGMENT, 635,      586, 504, 10, UNI;
VSGMENT, 636,      586, 587, 14, UNI;
VSGMENT, 637,      589, 473, 156, UNI;
VSGMENT, 638,      505, 510, 10, UNI;
VSGMENT, 639/TUR1A, 500, 501, 11, B1, 1;
VSGMENT, 640/TUR1B, 502, 503, 10, B1, 1;
VSGMENT, 641,      502, 504, 16, UNI;
VSGMENT, 642/TUR2A, 587, 588, 11, B1, 1;
VSGMENT, 643,      587, 589, 10, UNI;
VSGMENT, 644/TUR2B, 589, 590, 10, B1, 1;

```

```

VFLEET,AGV/1,25,4,4,3,3,5,8,2,5,28,
STOP(163,164,496,497),163(5,149)/164(10,158)/497(5,484)/
496(5,498);

```

```

CREATE A FULL LOAD AND ASSIGN TO A WORK STATION (LOAD
CAPACITY = 2)

```

```

CREATE,TRIAG(248,315,347,1),1,,80,1;

```

```

FIND AN EMPTY WORKSTATION

```

```

EVE1 EVENT,1,1;

```

```

ALL WORKSTATIONS ARE FULL, TRY AGAIN -- OR--

```

```

:      FOUND AN EMPTY WORKSTATION
:
:      ACTIVITY/1,1,ATRIB(6).NE.0,EVE1;NOLOAD
:      ACTIVITY,,ATRIB(6).EQ.0;
:      TERM;
:
:      WORKSTATIONS ONE THRU SIX
:
:      A LOAD ENTITY AND A MODULE ENTITY MUST BE PRESENT AT THE
:      WORKSTATION BEFORE AN AGV CAN BE CALLED.
:
:      QUE1 QUEUE(1),0,2,,LOD1;          Q1
:      QUE7 QUEUE(7),0,2,,LOD1;          \
:      LOD1 SELECT,ASM,,,QUE7,QUE1;      \__ ASM __ EVE2
:      ACTIVITY(1)/2,,,EVE2;REC1        /
:      QUE2 QUEUE(2),0,2,,LOD2;          /
:      QUE8 QUEUE(8),0,2,,LOD2;          Q7
:      LOD2 SELECT,ASM,,,QUE8,QUE2;
:      ACTIVITY(1)/3,,,EVE2;REC2
:      QUE3 QUEUE(3),0,2,,LOD3;
:      QUE9 QUEUE(9),0,2,,LOD3;
:      LOD3 SELECT,ASM,,,QUE9,QUE3;
:      ACTIVITY(1)/4,,,EVE2;REC3
:      QUE4 QUEUE(4),0,2,,LOD4;
:      QU10 QUEUE(10),0,2,,LOD4;
:      LOD4 SELECT,ASM,,,QU10,QUE4;
:      ACTIVITY(1)/5,,,EVE2;REC4
:      QUE5 QUEUE(5),0,2,,LOD5;
:      QU11 QUEUE(11),0,2,,LOD5;
:      LOD5 SELECT,ASM,,,QU11,QUE5;
:      ACTIVITY(1)/6,,,EVE2;REC5
:      QUE6 QUEUE(6),0,2,,LOD6;
:      QU12 QUEUE(12),0,2,,LOD6;
:      LOD6 SELECT,ASM,,,QU12,QUE6;
:      ACTIVITY(1)/7,,,EVE2;REC6
:
:      FIND AN EMPTY STAND IN THE MODULE STORAGE AREA
:      FIND A STAND? NO -- TRY AGAIN YES -- CALL AGV
:
:      EVE2 EVENT,2,1;
:      ACTIVITY/8,10,ATRIB(6).NE.0,EVE2;FINDMSA
:      ACTIVITY,,ATRIB(6).EQ.0,GO1;
:      GO1 GOON,1;
:      TERM;
:
:      ORDERS FOR AN AGV TO TAKE THE STOW MODULE TO STORAGE APPEAR
:      HERE
:
:      EN01 ENTER,1,1;
:
:      IF MODULE WAS NOT PRESENT WHEN LOAD WAS READY, LOAD IT
:
:      GOON,1;
:      ACTIVITY/9,22*ATRIB(4),ATRIB(3).LT.0,VW13;LOADNOTRDY

```

```

      ACTIVITY,,ATRIB(3).GE.0,VW13;
      CALL FOR AGV
VW13  VWAIT(13),AGV,ATRIB(1),CLOSEST,MATCH,1;RECTOMSA
      LOAD AGV
      ACTIVITY/18,30;RECMSA
      LEAVING?  FIND AN EMPTY MODULE AND TELL AN AGV TO BRING
      IT TO THE STAND YOU'RE LEAVING.
EVE3  EVENT,3,1;
      LEAVING RECEIVING SLOT.  RELEASE REC RESOURCE.
      FREE,ATRIB(7)/1,1;
      ATRIB(7) = MSA RESOURCE, FROM ATRIB = TO STORAGE ATRIB,
      START TIMER
      ASSIGN,ATRIB(7)=ATRIB(8),ATRIB(1)=ATRIB(2),ATRIB(9)=TNOW,
      ATRIB(5)=0;
      GO TO MODULE STORAGE STAND *?
      VMOVE,ATRIB(2),1;
      COLLECT TIME OF TRAVEL FROM RECEIVING TO MSA
      COLCT(1),INT(9),REC_TO_MSA,10/0/45,1;
      UNLOAD AGV
      ACTIVITY,30;
      VFREE,AGV,,,1;
      FQU IS THE QUEUE WHICH HOLDS ALL THE FULL MODULES
      UNTIL THEY ARE CALLED FOR BY THE CRANES IN THE HIGHRISE.
FQU  QUEUE(14);
      SENDING AN EMPTY MODULE TO RECEIVING
      AWA1 CONTAINS EMPTY MODULE ENTITIES PLACED THERE BY THE
      ORDER PROCESSOR.  THEY ARE WAITING FOR THE REC RESOURCE
      (RECEIVING SLOT.
EN02  ENTER,2,1;
AWA1  AWA1(15),ATRIB(8)/1,,1;
      CALL AGV
      VWAIT(16),AGV,ATRIB(1),CLOSEST,MATCH,1;MSATOREC

```

```

ACTIVITY/19,30;MSAREC
:
:   LEAVING?  RELEASE MSA RESOURCE (MSA STAND)
:
:   FREE,ATRIB(7)/1;
:
:   SECONDARY RESOURCE (RECEIVING STAND) BECOMES PRIMARY RESOURCE
:   FROM DESTINATION = TO DESTINATION START TIMER
:   ASSIGN, ATRIB(7)=ATRIB(8),ATRIB(1)=ATRIB(3),ATRIB(9)=TNOW;
:
:   TRAVEL TO RECEIVING
:
:   VMOVE,ATRIB(3),1;
:   COLLECT TIME TO TRAVEL FROM MSA TO RECEIVING
:   COLCT(2),INT(9),MSA_TO_REC,10/0/45,1;
:
:   UNLOAD AGV
:
:   ACTIVITY,30;
:   VFREE,AGV,,,1;
:
:   DEPOSIT EMPTY MODULE INTO THE PROPER RECEIVING QUEUE (*7-*12)
EVE4  EVENT,4,1;
      TERM;
:
:   EMPQ CONTAINS ALL THE EMPTY MODULES RESIDING IN THE MODULE
:   STORAGE AREA.
:
EMPQ  QUEUE(17);
:
:   HIGHRISE BIN AND RACK STORAGE BUILDING
:
:   AWA2 REPRESENTS ALL THE PAIRS OF STANDS FOR ROWS 1 THRU 60.
:   ALLOC(1) IS CALLED WHEN ANY OF THE RESOURCES *110 THRU 169
:   ARE AVAILABLE FOR ANY OF THE ENTITIES IN THE AWAIT NODE WITH
:   AN ATRIB(7) EQUAL TO THE RESOURCE NUMBER.
:
AWA2  AWAIT(18),ALLOC(1),,1;
      ACT,1;
:
:   EVENT 5 -- DETERMINE HOW MANY PICKS TO MAKE, TYPE OF PICKS TO MAKE
:   AND TOTAL PICKING AND STOWING TIME
:   GENERATE CALL FOR EMPTY OR STOW MODULE IF NEEDED
EVE5  EVENT,5,1;
:
:   PICKING AND STOWING ACTIVITY (ATRIB(2) + 40 SEC RETURN TO STAND)
:
ACTIVITY/10,ATRIB(2)+40,,COU;ASRS TIME
:
:   RECORD NUMBER OF PICKS IN INTERVAL (XX(95))
:   AND TOATL NUMBER OF PICKS (XX(96))

```

```

:
COU  ASSIGN, XX(95)=XX(95)+ATRIB(5),XX(96)=XX(96)+ATRIB(5);
:      WAIT UNTIL ORDER ARRIVES, OR PROCEED IF NO ORDER HAS BEEN PLACED
EVE7  EVENT,7,1;
      ACTIVITY/11,10,ATRIB(6).NE.0,EVE7; ORDNOTINYET
      ACTIVITY,,ATRIB(6).EQ.0; ORDER IS IN
:
:      FIND A STAND IN THE CONSOLIDATION AREA. IF ONE IS
:      FOUND, AN ORDER IS PLACED FOR AN AGV AND THE ENTITY COMES
:      OUT AT ENTER,3.
EVE8  EVENT,8,1;
      ACTIVITY/12,10,ATRIB(6).GT.0,EVE8;NOCONSTAND FOUND
      ACTIVITY,,ATRIB(6).LE.0;
      TERM;
:
:      TRANSPORT PICK MODULES TO THE CONSOLIDATION AREA
:
EN03  ENTER,3,1;
      GOON,1;
      ACTIVITY/22,,ATRIB(9).EQ.-1,EMPT;
      ACTIVITY/23,,ATRIB(9).NE.-1,VW19;
EMPT  EVENT,9,1;
      TERM;
VW19  VWAIT(19),AGV,ATRIB(1),CLOSEST,MATCH,1;MSATOCON
      ACTIVITY/20,30;HIGHCORB
:
:      DETERMINE IF ORDER FOR AN EMPTY OR STOW MODULE SHOULD BE MADE
EVE9  EVENT,9,1;
      ASSIGN, ATRIB(9)=TNOW,ATRIB(1)=ATRIB(3);
LEF   VMOVE,ATRIB(3),1;
:      COLLECT TIME FROM HIGHRISE TO CONSOLIDATION OR BUFFER
      GOON,1;
      ACTIVITY/13,,ATRIB(7).GT.211,BCOL;TOBUF
      ACTIVITY/14,,ATRIB(7).LE.211,CCOL;TOCON
BCOL  COLCT(3),INT(9),HIGH_TO_BUF,10/0/45,1;
      ACTIVITY,,,DIS1;
CCOL  COLCT(4),INT(9),HIGH_TO_CON,10/0/45,1;
      ACTIVITY,,,DIS1;
DIS1  GOON,1;
      ACTIVITY,30;
      VFREE,AGV,,,1;
:
:      IS THE PICK MODULE GOING TO THE CONSOLIDATION STAND OR THE
:      BUFFER STAND?
:
      ACTIVITY,,ATRIB(7).LE.211,CON;GOTO CONSOLIDATION
      ACTIVITY,,ATRIB(7).GT.211,BUF;GOTO BUFFER
:
:      BUFFER -- BUFFER AWAIT NODE GRABS THE CONSOLIDATION RESOURCES
:      SO THE ENTITIES IN THE BUFFER HAVE FIRST PRIORITY ON THE
:      CONSOLIDATION STANDS.
:
BUF   AWAIT(20),ALLOC(2),,1;
      TERM;

```

```

:
:      TAKE PICK MODULE IN BUFFER TO CONSOLIDATION
EN04  ENTER,4,1;
      VWAIT(21),AGV,ATRIB(1),CLOSEST,MATCH,1;BUFTOCON
      ACTIVITY/21,30;BUFCON
:
:      RELEASE BUFFER STAND
      FREE,ATRIB(7),1;
:      SET MODULE ATTRIBUTES
      ASSIGN,ATRIB(7)-ATRIB(8),ATRIB(9)=TNOW;
      VMOVE,ATRIB(3),1;
:      COLLECT TIME OF TRAVEL FROM BUF TO CON
      COLCT(5),INT(9),BUF_TO_CON,10/0/45,1;
:
:      UNLOAD AGV
:
      ACTIVITY,30;
      VFREE,AGV,,,1;
:
:      CONSOLIDATION UNPACKING ACTIVITY
CON   GOON,1;
EV10  EVENT,10,1;
      TERM;
:
      ENTER,5,1;
      ACTIVITY,ATRIB(5)*30;
:
:      CONQ HOLDS THE EMPTY MODULES TO FILL ORDERS FOR EMPTIES
:      IN THE HIGHRISE OR TO BE TAKEN BACK TO THE MSA.
CONQ  QUEUE(22);
:      ORDQ HOLDS ALL ORDERS FOR AGVs
ORDQ  QUEUE(23);
:
:      MSA TO HIGHRISE NETWORK
:
      ENTER,37,1;
      VWAIT(24),AGV,ATRIB(1),CLOSEST,MATCH,1;MSATOHIGH
      ACTIVITY/15,30;MSAHIGH
      FREE,ATRIB(7),1;
      ASSIGN,ATRIB(1)=ATRIB(3),ATRIB(7)=ATRIB(8),ATRIB(9)=TNOW;
      VMOVE,ATRIB(3),1;
      ACTIVITY,,,CO1;
CO1   COLCT(6),INT(9),MSA_TO_HIGH,10/0/45,1;
      VFREE,AGV,,,1;
      GOON,1;
      ACTIVITY,,,ATRIB(4).LT.0,END;
      ACTIVITY,,,AWA2;
:
:      CON EMPTY TO HIGHRISE NETWORK
:
      ENTER,38,1;
      VWAIT(25),AGV,ATRIB(1),CLOSEST,MATCH,1;CONTOHIGH
      ACTIVITY/16,30;CONHIGH
      FREE,ATRIB(7),1;

```

```

      ASSIGN, ATRIB(1)=ATRIB(3),ATRIB(7)=ATRIB(8),ATRIB(9)=TNOW;
V2    VMOVE,ATRIB(3),1;
      COLCT(7),INT(9),CON_TO_HIGH,10/0/45,1;
      VFREE,AGV,...,1;
      GOON,1;
      ACTIVITY,..,ATRIB(4).LT.0,END;
      ACTIVITY,..,AWA2;

      :
      :
      :
      CON EMPTY TO MSA NETWORK

      :
      :
      :
      ENTER,39,1;
      VWAIT(26),AGV,ATRIB(1),CLOSEST,MATCH,1;CONTOMSA
      ACTIVITY/17,30;CONMSA
      FREE,ATRIB(7),1;
      ASSIGN, ATRIB(1)=ATRIB(2),ATRIB(7)=ATRIB(8),ATRIB(9)=TNOW;
      VMOVE,ATRIB(2),1;
      COLCT(8),INT(9),CON_TO_MSA,10/0/45,1;
      VFREE,AGV,...,1;
      ACTIVITY,..,EMPQ;

      :
      :
      :
      HIGHRISE EMPTY TO MSA

      :
      :
      :
      ENTER,40,1;
VW29  VWAIT(29),AGV,ATRIB(1),CLOSEST,MATCH,1;HIGHTOMSA
      ACTIVITY/24,30;HIGHMSA
      EVENT,9,1;
      FREE,ATRIB(7),1;
      ASSIGN,ATRIB(1)=ATRIB(2),ATRIB(7)=ATRIB(8),ATRIB(9)=TNOW;
      VMOVE,ATRIB(3),1;
      COLCT(9),INT(9),HIGH_TO_MSA,10/0/45,1;
      ACTIVITY,30;
      VFREE,AGV,...,1;
      ACTIVITY,..,EMPQ;

      :
      :
      :
      STATISTIC TIMER -- COLLECTS STATISTICS ON PICKS MADE IN 15
      MINUTE INTERVALS AND CUMULATIVE COUNT.  ALSO DOES SAME
      FOR PICKS PROCESSED BY CONSOLIDATION

      :
      :
      :
      CREATE,..,1,1,1;
      ASSIGN,XX(130)=25;
EV13  EVENT,13,1;
      ACTIVITY,900,..,EV13;

      :
      :
      :
      ORDER PROCESSING TIMER
      CHECKS ORDERS EVERY 10 SECONDS

      :
      :
      :
      CREATE,..,1,1,1;
EV11  EVENT,11,1;
      ACTIVITY,10,..,EV11;

      :
      :
      :
      DUMMY AWAIT NODE -- RESOURCE TRANSACTIONS ARE DONE
      IN THE EVENT SUBROUTINES, BUT THE RESOURCES MUST BE ASSIGNED
      AN AWAIT NODE TO POLL.
DUMM  AWAIT(27),ATRIB(3),1;

```

```

END TERM;
:
:   PALLET MOVEMENT
:   CREATE PALLETS -- THE FIRST CREATE RUNS FOR THE FIRST
:   FOUR HOURS -- THE SECOND CREATE RUNS FOR THE NEXT THREE
:   HOURS
:
:   CREATE,271.7,0,,53,1;
:   ACT,,,P1;
:   CREATE,469.6,14400,,23,1;
:   ACT,,,P1;
:
:   CREATES FOUR PALLETS TO BE SENT TO SHIPPING
:   CREATE,3600,0,,4,1;
:   ASSIGN,TRIB(7)=4;
:   ACT,,,P4;
:   TRIB(7) IS THE TYPE OF PALLET
:       1 - LTL TO STORAGE
:       2 - LTL TO 3P&M
:       3 - 3P&M TO STORAGE
:       4 - STORAGE TO CONSOLIDATION
:
:   P1   GOON,1;
:   WHO GOES TO 3P&M?
:   ACTIVITY,,,05,P2;
:   ACTIVITY,,,95,P3;
:   P2   ASSIGN,TRIB(7)=2;
:   ACTIVITY,,,P4;
:   P3   ASSIGN,TRIB(7)=1;
:
:   EVENT DETERMINES WHEN TO SEND THE PALLET
:
:   P4   EVENT,12,1;
:   TERM;
:   ENTER,41,1;
:   VWAIT(30),AGV,TRIB(1),CLOSEST,MATCH,1;
:   ACTIVITY,30;
:   ASSIGN,TRIB(9)=TNOW;
:   GOON,1;
:
:   GO TO PROPER COLLECT NODE
:
:   ACTIVITY,,,TRIB(7).EQ.1,P5;
:   ACTIVITY,,,TRIB(7).EQ.2,P6;
:   ACTIVITY,,,TRIB(7).EQ.3,P7;
:   ACTIVITY,,,TRIB(7).GE.4,P8;
:   P5   GOON,1;
:   ACTIVITY,,,5,P5A;
:   ACTIVITY,,,5,P5B;
:   P5A  VMOVE,501,1;
:   VMOVE,503,1;
:   P5B  VMOVE,TRIB(3),1;
:   COLCT(10),INT(9),LTL_TO_STOR,10/0/45,1;
:   ACTIVITY,,,PD;

```



```

P6      VMOVE, ATRIB(3), 1;
        COLCT(11), INT(9), LTL_TO_3PM, 10/0/45, 1;
        ACT, 30;
        VFREE, AGV..., 1;
        ACTIVITY, TRIAG(1190, 1322, 1454, 10);
        ASSIGN, ATRIB(7)=3, ATRIB(1)=ATRI(3);
        ACTIVITY..., P4;
P7      GOON, 1;
        ACTIVITY..., .5, P7A;
        ACTIVITY..., .5, P7B;
P7A     VMOVE, 501, 1;
        VMOVE, 503, 1;
P7B     VMOVE, ATRIB(3), 1;
        COLCT(12), INT(9), PM_TO_STOR, 10/0/45, 1;
        ACTIVITY..., PD;
P8      GOON, 1;
        ACTIVITY..., .5, P8A;
        ACTIVITY..., .5, P8B;
P8A     VMOVE, 588, 1;
        VMOVE, 590, 1;
P8B     VMOVE, ATRIB(3), 1;
        COLCT(13), INT(9), STOR_TO_PCON, 10/0/45, 1;
        ACTIVITY..., PD;
PD      GOON, 1;
        ACTIVITY, 30;
        VFREE, AGV..., 1;
        TERM;

:
:      TROUBLESHOOTING LOOP
:

        CREATE, .5, 425200., .100, 1;
        ACT, ., EVQ;
        CREATE, .5, 136175., .100, 1;
EVQ     EVENT, 14, 1;
        TERM;
        ENDNETWORK;
SEEDS, +29104195799, + 3165019677, +34250765275,
        +83433676731, + 2043442035, +48027731405,
        +68148978141, +86533911893, +41946829757,
        +42229020695;
SIMULATE;
SEEDS, +23394537701, +31630593221, +57683330575,
        +63557341293, +19876448865, +37390217483,
        +96714363503, +75219541227, + 7775337281,
        +17566282657;
SIMULATE;
SEEDS, +83179952577, +87365015093, +90024513839,
        + 1896922103, +74734328227, +87358460117,
        +63520822907, +25754432491, +38132472385,
        +38803625967;
SIMULATE;
SEEDS, +79085984331, +88542545493, +43565217361,
        +49316758297, +20176679165, +80845733199,
        +77745828459, + 7049936053, + 346429173,

```

+80327117303;  
SIMULATE;  
SEEDS,+75745780337,+85964712537,+79456160067,  
+16377472033,+71580277133,+19855390197,  
+98634918501,+ 3512720831,+67545066329,  
+45122139197;  
SIMULATE;  
SEEDS,+33991132813,+43523526777,+22301436363,  
+12295019709,+69073217513,+96372876779,  
+57712868009,+66307487977,+86080821533,  
+19844759491;  
SIMULATE;  
SEEDS,+84430429477,+95518786117,+75825920211,  
+77942837001,+46229315295,+49488735669,  
+66442951041,+18038610259,+83094128917,  
+25500274911;  
SIMULATE;  
SEEDS,+70850793161,+16769878253,+13055119099,  
+23833129905,+52962890591,+94689290945,  
+32774410007,+71787489301,+12648419233,  
+99662595945;  
SIMULATE;  
SEEDS,+36881465085,+16457072901,+89443897611,  
+59017478943,+56059044683,+86252793955,  
+85012641359,+ 441597887,+13879757409,  
+21176115219;  
SIMULATE;  
SEEDS,+ 4357361469,+41630910917,+ 2391045255,  
+71842698761,+15779916277,+48688998041,  
+76451962469,+67255249511,+29592847213,  
+63274076565;  
SIMULATE;  
FIN;

Appendix B. Additional Fortran Code

```
PROGRAM MAIN
DIMENSION NSET(400000)
INCLUDE 'PARAM.INC'
INTEGER EMPTY(1:60),IPG(1:3,1:60),DELIV(1:60)
COMMON/SCOM1/ATRIB(MATRB), DD(MEQT), DDL(MEQT), DTNOW, II, MFA,
1MSTOP,NCLNR, NCRDR, NPRNT, NNRUN, NNSET, NTAPE, SS(MEQT),
2SSL(MEQT),TNEXT, TNOW, XX(MMXXV)
COMMON QSET(400000)
COMMON/UCOM1/ IPG, EMPTY,DELIV
EQUIVALENCE (NSET(1),QSET(1))
NNSET = 400000
NCRDH=5
NPRNT=6
NTAPE=7
OPEN(UNIT=9,FILE='COL123.OUT',STATUS='NEW')
OPEN(UNIT=10,FILE='EFF123.OUT',STATUS='NEW')
OPEN(UNIT=11,FILE='MSA123.OUT',STATUS='NEW')
OPEN(UNIT=12,FILE='CON123.OUT',STATUS='NEW')
CALL SLAM
CLOSE(9)
CLOSE(10)
CLOSE(11)
CLOSE(12)
STOP
END
```

```
SUBROUTINE INTLC
INCLUDE 'PARAM.INC'
COMMON/SCOM1/ATRIB(MATRB), DD(MEQT), DDL(MEQT), DTNOW, II, MFA,
1MSTOP,NCLNR, NCRDR, NPRNT, NNRUN, NNSET, NTAPE, SS(MEQT),
2SSL(MEQT),TNEXT, TNOW, XX(MMXXV)
*      IPG -- NUMBER OF TYPE 1, 11, AND 111 PICKS PER ROW.
*      MPTMODS -- INITIAL NUMBER OF EMPTY MODS IN THE MSA.
INTEGER MPTMODS, EMPTY(1:60),IPG(1:3,1:60),TEMP,DELIV(1:60)
REAL A(17),B(17),C(17),IPG1,IPG2,IPG3
COMMON/UCOM1/ IPG, EMPTY,DELIV
MPTMODS=74
* STATISTICS COUNTERS
XX(121)=-1
XX(123)=-1
*
*      FILL EMPTY QUEUE WITH EMPTY MODULES
*
DO 10 I=1,MPTMODS
*
*      MSA DESTINATION CONTROL POINT
A(1)=GETARY(1,1)
A(2)=A(1)
*      NO STOWS (EMPTY MODULE)
A(4)=0
*      NO PICKS
A(5)=0
```

```

*       SET FLAG
*       A(6)=0

*       RESOURCE NUMBER ASSOCIATED WITH MSA SLOT
*       A(7)=1
*       GRAB SLOT RESOURCE (SIGNIFIES THE SLOT IS TAKEN)
*       CALL SEIZE(I,1)
*       FILE EMPTY MODULE IN EMPTY QUEUE
*       CALL FILEM(17,A)
10  CONTINUE
*
*
20  DO 30 I=1,6
*
*       FOR RECEIVING SLOTS 1,3,5,7,9,11
*       RECEIVING CONTROL POINTS
*       B(1)=(GETARY(4,((2*I)-1)))
*       B(3)=B(1)
*       SET FLAG
*       B(6)=0
*       RESOURCE NUMBER ASSOCIATED WITH RECEIVING SLOT
*       RESOURCES #240 THRU #251 BY 2
*       B(7)=239+((2*I)-1)
*       IU-B(7)
*       CALL SEIZE(IU,1)
*       FILE EMPTY MODULE IN RECEIVING QUEUE #7 - #12
*       CALL FILEM(6+1,B)
*
*       SAME AS ABOVE EXCEPT FOR RECEIVING SLOTS 2,4,6,8,10,12
*       B(1)=GETARY(4,2*I)
*       B(3)=B(1)
*       B(7)=239+(2*I)
*       IU-B(7)
*       CALL SEIZE(IU,1)
*       CALL FILEM(6+1,B)
30  CONTINUE
*
*       PLACE AN EMPTY MODULE AT EACH HIGHRISE ROW
*
*       II=0
40  DO 50 I=1,60
*       ROW CONTROL POINT
*       C(1)=GETARY(2,1)
*       C(3)=C(1)
*       NO STOWS ON BOARD
*       C(4)=0
*       SET FLAG
*       C(6)=0
*       ASSIGN RESOURCE ASSOCIATED WITH THE SLOT (#110 - #169)
*       C(7)=1+109
*       FILE EMPTY MODULE IN PROPER AWAIT NODE
*       CALL FILEM(18,C)
*       DETERMINE HOW MANY IPG 1, 11, AND 111 PICKS WILL BE MADE
*       IN A ROW
*       IPG1=RNORM(34.,7.,2)

```

```

      IPG(1,1)=(IPG1)+.5
      IPG2=RNORM(67.,10.,2)
      IPG(2,1)=(IPG2)+.5
      IPG3=RNORM(158.,16.,2)
      IPG(3,1)=(IPG3)+.5
*      CAN'T HAVE NEGATIVE PICKS NOW CAN WE???
      DO 45,J=1,3
        IF (IPG(J,1).LE.0) THEN
          IPG(J,1)=0
        ENDIF
*      II -- TOTAL NUMBER OF PICKS TO BE PICKED
        II=II+IPG(J,1)
45      CONTINUE
        EMPTY(1)=0
*
50      CONTINUE
*      CLEAR COUNTERS
60      DO 70 I=1,100
        XX(I)=0
70      CONTINUE
        XX(94)=1
        RETURN
      END

      SUBROUTINE EVENT(I)
      INCLUDE 'PARAM.INC'
      COMMON/SCOM1/ATRIB(MATRB), DD(MEQT), DDL(MEQT), DTNOW, II, MFA,
      IMSTOP,NCLNR, NCRDR, NPRNT, NNRUN, NNSET, NTAPE, SS(MEQT),
      2SSL(MEQT),TNEXT, TNOW, XX(MMXXV)
      REAL FSTOW(2),STOW(2),FPICK(2),PICK(2),A(17),B(17),XVAL
      INTEGER MSASTAN, TEMP, IPG(1:3,1:60), EMPTY(1:60),DELIV(1:60)
      INTEGER NRANKQ,11,12,13
      COMMON/UCOM1/ IPG, EMPTY,DELIV
*      CUMULATIVE DISTRIBUTIONS FOR STOW TIMES
      DATA FSTOW/.915,1./
*      STOW TIMES
      DATA STOW/115.,188./
*      CUMULATIVE DISTRIBUTIONS FOR PICK TIMES
      DATA FPICK/.9,1./
*      PICK TIMES
      DATA PICK/66.,107./
*      NUMBER OF MODULE STORAGE STANDS
      MSASTAN = 74
*
*      EVENTS 1 THRU 6 -- 6 IS NOT USED -- IF YOU HAVE AN EVENT
*      USE EVENT 6 AND CHANGED THE SECOND 60 IN THE GOTO
      IF (I.LE.6) THEN
        GOTO (10,20,40,50,60,60) I
      ENDIF
*      EVENTS 7 THRU 14
      GOTO (100,110,130,135,140,200,230,400) I-6
*      EVENT 1 -- PICK WHICH RECEIVING LANE WILL GET THE LOAD
*
*      PRODUCES UNIFORM INTEGERS FROM 1 TO 6

```

```

010 J=(UNFRM(.5,6.499,3))+.5
*   DON'T SEND A NEW LOAD TO THE LAST DESTINATION
  ATRIB(6)=0
  IF (XX(94).EQ.J) THEN
    GOTO 10
  ENDIF
*   IF RECEIVING THE RECEIVING LANE IS FULL, TRY ANOTHER SLOT
  IF (NNQ(J).GT.1) THEN
    ATRIB(6)=1
    GOTO 15
  ENDIF
*   FILE THE LOAD IN THE PROPER RECEIVING LANE
  CALL FILEM(J,ATRIB)
*   UPDATE THE COUNTER WITH THE LANE THE LOAD WAS SENT TO
  XX(94)=J
015 RETURN
*
*   EVENT 2 -- FIND EMPTY SLOT FOR FULL RECEIVING MODULE
*
*   SET FLAG -- IF ENTITY RETURNS WITH FLAG = 1, EVENT 2 IS REPEATED
*
020 ATRIB(6)=1
*   CHECK FOR EMPTY MSA STAND
*   RESOURCE 73 IS A STAND THAT HANDLES ALL THE OVERFLOW
*   IT IS NORMALLY CAPABLE OF HANDLING 25 MODULES
DO 030 J=1,MSASTAN-1
*   FOUND AN EMPTY SLOT? YES -- GRAB IT NO -- TRY AGAIN
  IF (NNRSC(J).GT.0) THEN
    CALL SEIZE(J,1)
    SET DESTINATION
    ATRIB(2) = GETARY(1,J)
    RECORD RESOURCE
    ASSIGN NUMBER OF STOWS
    TEMP=(UNFRM(19.5,30.499,4))+.5
    ATRIB(4)=TEMP
    NO PICKS
    ATRIB(5)=0
    ATRIB(10)=TNOW
*   ASSIGN HIGHRISE DESTINATION
    K=(UNFRM(.5,60.499,4))+.5
    IF (K.LE.0) THEN
      K=1
    ENDIF
    IF (K.GE.60) THEN
      K=60
    ENDIF
    ATRIB(3)=GETARY(2,K)
*   ATRIB(8) HAS RESOURCE OF DESTINATION
    ATRIB(8) = J
*   ORDER TYPE - 1 SEND AGV TO RECEIVING TO PICK UP STOW MODULE
    ATRIB(6)=1
    PLACE ORDER
    CALL FILEM(23,ATRIB)
*   SET FLAG TO 0

```

```

        ATRIB(6) = 0
        RETURN
    ENDIF
030  CONTINUE
    RETURN

*
*      EVENT 3 -- RECEIVING CALL FOR EMPTY MODULE
*
*      CLEAR TEMPORARY ARRAY
040  DO 45 J=1,8
        A(J)=ATRIB(J)
045  CONTINUE
*      TO DESTINATION - FROM DESTINATION
        A(3)=ATRIB(1)
*      ASSIGN SECOND RESOURCE
        A(8)=ATRIB(7)
*      ORDER CODE = 2 -- BRING EMPTY TO RECEIVING
        A(6) = 2
*      SEND ORDER
        CALL FILEM (23,A)
        RETURN
*
*      EVENT 4 -- PLACES EMPTY MODULE SENT TO RECEIVING IN THE PROPER
*      QUEUE
*
*      TEMP IS AN INTEGER BETWEEN 7 AND 12
050  TEMP = ((ATRIB(3)/3)+6)
*
*      IF THERE IS A LOAD WAITING AND NO MODULE, THEN YOU NEED
*      TO TAKE THE TIME TO FILL IT.
*
*      LANE QUEUE          LANE MODULE QUEUE
        IF ((NNQ(TEMP-6).GT.0).AND.(NNQ(TEMP).EQ.0)) THEN
*          ATRIB 3 TELLS ACTIVITY TO LOAD THE MODULE IF NEGATIVE
            ATRIB(3)=-1
        ENDIF
*      FILE ENTITY IN QUEUE
        CALL FILEM(TEMP,ATRIB)
        RETURN
*
*
*      EVENT 5 -- DETERMINE THE NUMBER OF PICKS AND STOWS TO BE MADE
*      IN THE HIGHRISE
*
*      IF NO STOWS, THEN PICK A FULL LOAD -- IF STOWS, PICK A HALF
*      LOAD
060  IF (ATRIB(4).LE.0) THEN
        TEMP=(UNFRM(39.5,60.499,5))+.5
        ATRIB(5)=TEMP
    ELSE
        TEMP=(UNFRM(19.5,30.499,5))+.5
        ATRIB(5)=TEMP
    
```

```

ENDIF
*   J = ROW NUMBER
*   J = ATRIB(7)-109
*   DETERMINE IPG TYPE TO PICK
*
*   ANY TYPE 1s YES -- PICK THEM NO -- TRY TYPE 11s
IF (IPG(1,J).GT.0) THEN
  IF (ATRIB(5).LE.IPG(1,J)) THEN
    IPG(1,J)=IPG(1,J)-ATRIB(5)
  ELSE
    IF ALL THE IPG 1's ARE TO BE PICKED FIRST USE THIS SECTION
    START
      ATRIB(5) = IPG(1,J)
      IPG(1,J) = 0
    FINISH
  IF A COMBINED RUN OF TYPE 1 AND 11 PICKS IS TO BE MADE,
  COMMENT OUT THE PREVIOUS TWO LINES AND UNCOMMENT THE FOLLOWING:
  START OF SECTION TO DO 1/2 PICKS
  IF (IPG(2,J).GT.0) THEN
    IF ((ATRIB(5)-IPG(1,J)).LE.IPG(2,J)) THEN
      IPG(2,J)=IPG(2,J)-(ATRIB(5)-IPG(1,J))
      IPG(1,J)=0
    ELSE
      ATRIB(5)=IPG(1,J)+IPG(2,J)
      IPG(1,J)=0
      IPG(2,J)=0
    ENDIF
  ENDIF
  END OF SECTION TO DO 1/2 PICKS
ENDIF
ELSE
  CHECK FOR TYPE 11s IF NONE AVAILABLE, CHECK TYPE 111s
  IF (IPG(2,J).GT.0) THEN
    IF (ATRIB(5).LE.IPG(2,J)) THEN
      IPG(2,J)=IPG(2,J)-ATRIB(5)
    ELSE
      ATRIB(5)=IPG(2,J)
      IPG(2,J)=0
    ENDIF
  ELSE
    CHECK FOR TYPE 111 PICKS
    IF (IPG(3,J).GT.0) THEN
      IF (ATRIB(5).GT.IPG(3,J)) THEN
        ATRIB(5)=IPG(3,J)
        IPG(3,J)=0
      ELSE
        IPG(3,J)=IPG(3,J)-ATRIB(5)
      ENDIF
    ELSE
      ATRIB(5)=0
    ENDIF
  ENDIF

```



```

        ENDIF
    ENDIF
    ATRIB(2)=0
    *       DETERMINE TOTAL STOWING TIME
    DO 70 J=1,ATRIB(4)
        ATRIB(2)=ATRIB(2)+DPROB(FSTOW,STOW,2,1)
070  CONTINUE
    *       DETERMINE TOTAL PICKING TIME
    DO 80 J=1,ATRIB(5)
        ATRIB(2)=ATRIB(2)+DPROB(FPICK,PICK,2,1)
080  CONTINUE
    *
    *       -- GENERATE CALL FOR EMPTY OR STOW MODULE TO THE HIGHRISE
    *
    *
090  J=ATRIB(7)-109
    *       XX(J)=0 -- NO ORDERS PENDING
    XX(J)=0
    IU=ATRIB(7)
    *       ONLY PLACE ORDER IF NRUSE(IU).EQ.1 AND XX(J)=0
    *       (NO ORDER ON RECORD)
    IF ((NRUSE(IU).EQ.1).AND.(XX(J).EQ.0)) THEN
    *       3 -- CODE FOR SEND AN EMPTY OR STOW MODULE TO THE HIGHRISE
        ATRIB(6)=3
        CALL FILEM(23,ATRIB)
        XX(J)=1
    ENDIF
    RETURN

    *
    *       EVENT 7 -- WAIT UNTIL ORDER ARRIVES, OR PROCEED IF NO ORDERS
    *       HAVE BEEN PLACED
    *
100  ATRIB(6)=1
    *       CURRENT RESOURCE NUMBER
    IU=ATRIB(7)
    *       ROW NUMBER
    J=ATRIB(7)-109
    *       CURRENT CONTROL POINT
    T=ATRIB(3)
    NRANKQ=NFIND(1,18,3,0,T,0.)
    *       IF NRANKQ.GT.0 THE ORDER IS IN -- IF DELIV(J).EQ.1
    *       THERE IS NO ORDER COMING, SO PUT THE MODULE IN THE
    *       PICKUP STAND WITHOUT WORRYING ABOUT A MODULE
    *       BEING DELIVERED TO THE DELIVERY STAND
    *
    IF ((NRANKQ.GT.0).OR.(DELIV(J).EQ.1)) THEN
    *       ADD ANOTHER UNIT TO RESOURCE ATRIB(7) SO THE ORDER CAN PROCEED
        CALL ALTER(IU,1)
        ATRIB(6)=0
    ENDIF
    RETURN

    *
    *       EVENT 8 -- FIND A STAND IN THE CONSOLIDATION AREA
    *

```

```

110  ATRIB(6)=1
    IF (ATRI(5).GT.0) THEN
*      FIND A STAND
      DO 120 J=180,221
        IF (NNRSC(J).GT.0) THEN
          CALL SEIZE(J,1)
          SET RESOURCE
          ATRIB(8)=J
          NEW DESTINATION
          ATRIB(3)=GETARY(3,J-179)
          ORDER = 4
          ATRIB(6)=4
          CALL FILEM(23,ATRI(6))
          SET FLAG
          ATRIB(6)=0
          RETURN
        ENDIF
120  CONTINUE
      RETURN
*      IF THE MODULE DOES NOT CONTAIN ANY PICKS, DO NOT SEND
*      IT TO THE CONSOLIDATION STANDS. IF THIS IS THE FIRST
*      EMPTY TO COME OUT OF THAT ROW, KEEP IT AT THE DELIVERY
*      STAND (THE ENTITY IS TERMINATED). IF THE EMPTY MODULE
*      IS NOT THE FIRST, SEND IT BACK TO THE MODULE STORAGE AREA
      ELSE
        IU=ATRI(7)-109
        IF FIRST EMPTY KEEP
        IF (EMPTY(IU).EQ.0) THEN
          EMPTY(IU)=1
          ATRIB(6)=4
          ATRIB(9)=-1
          CALL FILEM(23,ATRI(6))
          ATRIB(6)=0
          RETURN
        ELSE
          SEND BACK TO MSA
          DO 125 J=1,72
            IF (NNRSC(J).GT.0) THEN
*              SEIZE MSA STAND
              CALL SEIZE(J,1)
              ATRIB(8)=J
              ATRIB(2)=GETARY(1,J)
              MAKE ORDER FOR AGV
              ATRIB(6)=7
              CALL FILEM(23,ATRI(6))
              ATRIB(6)=0
              RETURN
            ENDIF
*          RESOURCE 74 IS AN OVERFLOW STAND FOR EMPTY
*          MODULES -- ITS SIZE IS NORMALLY 25
            IF (NNRSC(74).GT.0) THEN
              CALL SEIZE(74,1)
              ATRIB(8)=74
              ATRIB(2)=GETARY(1,74)

```

```

        ATRIB(6)=7
        CALL FILEM(23,ATRIB)
        ATRIB(6)=0
        RETURN
    ENDIF
125    CONTINUE
    ENDIF
    ENDIF
    RETURN
*
*    EVENT 9 -- RELEASE ASRS CRANE BY REDUCING RESOURCE(ATRIB(7)) BY 1
*    IF AN ORDER IS NEEDED, THEN DO THAT TOO
*
130    IU=ATRIB(7)
    CALL FREE(IU,1)
    CALL ALTER(IU,-1)
*    MAKE ORDER FOR EMPTY MODULE, IF NEEDED
*
*    IF XX EQUALS 0 (NO ORDER) AND NNRUSE - 1 (NO ORDER IN THE AWAIT
*    NODE) THEN MAKE AN ORDER
*
    J=ATRIB(7)-109
    IF ((XX(J).EQ.0).AND.(NRUSE(IU).EQ.1)) THEN
*        3 -- CODE FOR SEND AN EMPTY OR STOW MODULE TO THE HIGHRISE
        ATRIB(6)=3
        CALL FILEM(23,ATRIB)
        XX(J)=1
    ENDIF
    ATRIB(7)=ATRIB(8)
    RETURN
*
*
*    EVENT 10 -- COUNT THE NUMBER OF PICKS IN THE MODULE GOING
*    TO A CONSOLIDATION STAND
*
*
*    INDIVIDUAL CONSOLIDATION STAND COUNT
135    XX(ATRIB(7)-119)=XX(ATRIB(7)-119)+ATRIB(5)
*    TOTAL CONSOLIDATION BOX COUNT
    XX(93)=XX(93)+ATRIB(5)
*    INTERVAL CONSOLIDATION BOX COUNT
    XX(97)=XX(97)+ATRIB(5)
    CALL ENTER(5,ATRIB)
    RETURN
*
*
*    EVENT 11 -- ORDER PRE PROCESSOR -- ALL ORDERS FOR EMPTY MODULES,
*    FULL MODULES, OR TO REMOVE A MODULE ARE SORTED OUT HERE FIRST
*    BEFORE BEING SENT TO A VWAIT NODE. THIS EVENT IS CALLED EVERY
*    SO MANY SECONDS. IF AN ORDER CANNOT BE FILLED (i.e. NO EMPTIES
*    ARE AVAILABLE), IT WILL BE PUT BACK IN THE ORDER QUEUE.
*
*
*    ATRIB(6) CARRIES THE KEY TO THE TYPE OF ORDER.
*
*
*    1 = AGV NEEDED TO PICK UP STOW MODULE AT RECEIVING TO TAKE TO MSA
*    2 = AGV NEEDED TO BRING EMPTY MODULE FROM MSA TO RECEIVING

```

```

*      3 = AGV NEEDED TO BRING EMPTY OR STOW MODULE TO THE HIGHRISE
*      4 = AGV NEEDED TO TAKE PICK MODULE FROM HIGHRISE TO CONSOLIDATION
*      5 = AGV NEEDED TO TAKE EMPTY OR STOW MODULE TO HIGHRISE
*      6 = AGV NEEDED TO TAKE A PICK MODULE FROM BUF TO CON
*      7 = AGV NEEDED TO TAKE EMPTY MODULE FROM HIGHRISE TO MSA
*      8 = AGV NEEDED TO PICK UP PALLET AND DELIVER
*
*      .F NO ORDERS, GOTO 182
140  K=NNQ(23)
      IF (K.LE.0) THEN
          GOTO 182
      ENDIF
*      DO FOR EACH ORDER IN THE ORDER QUEUE
DO 180 J=1,K
*      REMOVE THE FIRST ORDER
      CALL RMOVE(1,23,A)
*      DETERMINE ORDER TYPE
      GOTO (145,150,155,160,165,170,175,178) A(6)
*      ORDER = 1
*      EXECUTE ORDER
145  CALL ENTER(1,A)
      GOTO 180
*
*      ORDER = 2
*      EMPTY MODULE AVAILABLE? YES -- GRAB IT  NO -- REFILE ORDER
150  IF (NNQ(17).GT.0) THEN
      CALL RMOVE(1,17,B)
*      ASSIGN TO DESTINATION
      B(3)=A(3)
*      ASSIGN SECOND RESOURCE (RECEIVING STAND)
      B(8)=A(7)
*      EXECUTE ORDER
      CALL ENTER(2,B)
      GOTO 180
      ELSE
*      NO LUCK
      CALL FILEM(23,A)
      ENDIF
      GOTO 180
*
*      ORDER = 3
*
155  L=A(7)-109
      DELIV(L)=0
      IF ((IPG(1,L).GT.0).OR.(IPG(2,L).GT.0)) THEN
*      IF EMPTY MODULE IN CONSOLIDATION, TAKE IT FIRST
156  IF (NNQ(22).GT.0) THEN
          CALL RMOVE(1,22,B)
          B(3)=A(1)
          B(8)=A(7)
          B(4)=0
*      EXECUTE ORDER
          CALL ENTER(38,B)
          GOTO 180

```

```

ELSE
*      CHECK MSA FOR EMPTY MODULES
      IF (NNQ(17).GT.0) THEN
        CALL RMOVE(1,17,B)
        B(3)=A(1)
        B(8)=A(7)
        B(4)=0
        CALL ENTER(37,B)
        GOTO 180
      ELSE
*      NO LUCK REFILE ORDER
        CALL FILEM(23,A)
        GOTO 180
      ENDIF
ENDIF
ELSE
*      IF ONLY TYPE IIIs LEFT, CHECK FOR STOW MODULE FIRST, THEN
*      EMPTY MODULE
*      START OF SECTION TO DO PICK/STOW
      IF (IPG(3,L).GE.0) THEN
        XVAL=A(1)
        NRANKQ=NFIND(1,14,3,0,XVAL,0.0)
        IF (A(10).EQ.0) THEN
          A(10)=1
        ENDIF
*      IF A STOW MODULE IS FOUND, GRAB IT
        IF (NRANKQ.GT.0) THEN
          CALL RMOVE(NRANKQ,14,B)
          B(8)=A(7)
          CALL ENTER(37,B)
          GOTO 180
        ELSE
*      IF THERE ARE NOT ANY TYPE IIIs LEFT OR ANY STOW
*      MODULES, REFILE THE ORDER -- THE REFILED ORDER
*      KEEPS CYCLING UNTIL THE END OF THE SIMULATION
*      OR A STOW MODULE IS CREATED
          IF (IPG(3,L).LE.0) THEN
            CALL FILEM(23,A)
            DELIV(L)=1
            GOTO 180
          ENDIF
*      IF THERE ARE TYPE IIIs LEFT, BUT NO STOWS, FIND AN EMPTY
          IF (IPG(3,L).GT.0) THEN
*      CHECK THE CONSOLIDATION AREA FOR AN EMPTY MODULE
            IF (NNQ(22).GT.0) THEN
              CALL RMOVE(1,22,B)
              B(3)=A(1)
              B(8)=A(7)
              B(4)=0
*      EXECUTE ORDER
              CALL ENTER(38,B)
              GOTO 180
*      TRY THE MODULE STORAGE AREA FOR AN EMPTY
            ELSE

```

```

*           CHECK MSA FOR EMPTY MODULES
IF (NNQ(17).GT.0) THEN
    CALL RMOVE(1,17,B)
    B(3)=A(1)
    B(8)=A(7)
    B(4)=0
    CALL ENTER(37,B)
    GOTO 180
ELSE
    NO LUCK REFILE ORDER
    CALL FILEM(23,A)
    DELIV(L)=1
    GOTO 180
ENDIF
ENDIF
ENDIF
END OF SECTION TO DO PICK AND STOWS

*
* START OF PICK ALL THEN STOW
* IF SEPARATE PICK RUNS AND STOW RUNS ARE TO BE MADE, USE THIS SECTION
* AND COMMENT OUT THE PICK AND STOW SECTION
*   IF (IPG(3,L).GE.0) THEN
*       XVAL=A(1)
*       NRANKQ=NFIND(1,14,3,0,XVAL,0.0)
*       IF (A(10).EQ.0) THEN
*           A(10)=1
*       ENDIF
*       IF ((NRANKQ.GT.0).AND.(IPG(3,L).LE.0)) THEN
*           CALL RMOVE(NRANKQ,14,B)
*           B(8)=A(7)
*           CALL ENTER(37,B)
*           GOTO 180
*       ELSE
*           IF (IPG(3,L).LE.0) THEN
*               CALL FILEM(23,A)
*               DELIV(L)=1
*               GOTO 180
*           ENDIF
*           IF (IPG(3,L).GT.0) THEN
*               IF (NNQ(22).GT.0) THEN
*                   CALL RMOVE(1,22,B)
*                   B(3)=A(1)
*                   B(8)=A(7)
*                   B(4)=0
*                   EXECUTE ORDER
*                   CALL ENTER(38,B)
*                   GOTO 180
*               ELSE
*                   CHECK MSA FOR EMPTY MODULES
*                   IF (NNQ(17).GT.0) THEN
*                       CALL RMOVE(1,17,B)

```

```

*           B(3)=A(1)
*           B(8)=A(7)
*           B(4)=0
*           CALL ENTER(37,B)
*           GOTO 180
*
*           ELSE
*
*               NO LUCK REFILE ORDER
*               CALL FILEM(23,A)
*               DELIV(L)=1
*               GOTO 180
*
*           ENDIF
*       ENDIF
*   ENDIF
*   ENDIF
*   FINISH OF PICK ALL FIRST THEN STOW
ENDIF
GOTO 180

```

```

*
*       ORDER = 4
*
*       EXECUTE ORDER
160  CALL ENTER(3,A)
      GOTO 180
*
*       ORDER = 5
*
*       EXECUTE ORDER
165  CALL ENTER(37,A)
      GOTO 180
*
*       ORDER = 6
*
*       EXECUTE ORDER
170  CALL ENTER(4,A)
      GOTO 180
*       ORDER = 7
175  CALL ENTER(40,A)
      GOTO 180
*       ORDER = 8
178  CALL ENTER(41,A)
      GOTO 180
180  CONTINUE
*
*       IF ANY EMPTY MODULES REMAIN IN THE CONSOLIDATION AREA,
*       TAKE THEM TO THE MSA
*
182  IF (NNQ(22).GT.0) THEN
      K=NNQ(22)
      DO 190 J=1,K
          CALL RMOVE(1,22,A)
          DO 185 L=1,72
              IF (NNKSC(L).GT.0) THEN
                  CALL SEIZE(L,1)

```

```

        A(8)=L
        A(2)=GETARY(1,L)
        A(4)=0
        CALL ENTER(39,A)
        GOTO 190
    ENDIF
185    CONTINUE
*    IF THERE AREN'T ANY STANDS AVAILABLE, PUT THE EMPTIES INTO
*    OVERFLOW.
        IF (NNRSC(74).GT.0) THEN
            CALL SEIZE(74,1)
            A(8)=74
            A(2)=GETARY(1,74)
            A(4)=0
            CALL ENTER(39,A)
        ELSE
            CALL FILEM(22,A)
        ENDIF
190    CONTINUE
    ENDIF
    RETURN

*
*    EVENT 12 -- CONTROLS THE MOVEMENT OF PALLETS
*
*    XX(101) IS THE COUNTER FOR THE SIX LTL STANDS
*    IT CYCLES FROM 1 THRU 6
200    XX(101)=XX(101)+1
        IF (XX(101).EQ.7) THEN
            XX(101)=1
        ENDIF
*    ATRIB 7 IS THE ROUTE THE PALLET WILL TRAVEL
        GOTO (202,204,206,208) ATRIB(7)
*    LTL TO PALLET STORAGE
202    L=XX(101)
*    SET LOCATION IN LTL
        ATRIB(1)=GETARY(5,L)
*    ASSIGN DESTINATION IN PALLET STORAGE
        K=UNFRM(.5,29.499,9)+.5
        ATRIB(3)=GETARY(7,K)
*    ASSIGN TYPE OF ORDER
        ATRIB(6)=8
*    FILE ORDER
        CALL FILEM(23,ATRIB)
        RETURN
*    LTL TO 3PM
204    L=XX(101)
*    ASSIGN LTL DESTINATION
        ATRIB(1)=GETARY(5,L)
*    ASSIGN 3P&M DESTINATION
        K=UNFRM(.5,9.499,9)+.5
        ATRIB(3)=GETARY(6,K)
*    ASSIGN TYPE OF ORDER
        ATRIB(6)=8
*    FILE ORDER

```



```

      CALL FILEM(23,ATTRIB)
      RETURN
*       3PM TO PALLET STORAGE
*       ASSIGN PALLET STORAGE DESTINATION
206 K=UNFRM(.5,29.499,9)+.5
      ATTRIB(3)=GETARY(7,K)
*       ASSIGN TYPE OF ORDER
      ATTRIB(6)=8
*       FILE ORDER
      CALL FILEM(23,ATTRIB)
      RETURN
*       PALLET STORAGE TO PALLET CONSOLIDATION
*       ASSIGN LOCATION IN PALLET STORAGE TO NEWLY CREATED PALLET
208 K=UNFRM(.5,29.499,9)+.5
      ATTRIB(1)=GETARY(7,K)
*       ASSIGN DESTINATION IN CONSOLIDATION
      ATTRIB(3)=480
*       ASSIGN TYPE OF ORDER
      ATTRIB(6)=8
*       FILE ORDER
      CALL FILEM(23,ATTRIB)
      RETURN
*
*       EVENT 13 -- STATISTICS
*       WRITE TO FILE 10 -- TNOW IN MINUTES, INTERVAL AND CUMULATIVE
*       ROW PICKS, AND INTERVAL AND CUMULATIVE CONSOLIDATION PICKS
230 WRITE(10,*) ,TNOW/60,XX(95),XX(96),XX(97),XX(93)
*       EFFIC COLLECTS EFFICIENCY STATISTICS ON THE AGV FLEET
      CALL EFFIC
*       LOOP TO CHECK THE OCCUPANCY RATE IN THE MSA -- 1 THRU
*       72 ARE SINGLE STANDS -- 73 IS AN OVERFLOW STAND FOR
*       STOW MODULES -- STAND 74 IS THE OVERFLOW STAND FOR
*       EMPTY MODULES --
      ISTAND=0
      DO 235 K=1,72
        ISTAND=ISTAND+NRUSE(K)
235 CONTINUE
*       WRITE TO FILE 10 -- TNOW IN MINUTES, NUMBER OF FULL STANDS
*       IN 1 THRU 72, NUMBER OF STOW MODULES IN 73, AND NUMBER
*       OF EMPTUES IN STAND 74
      WRITE(11,*) ,TNOW/60,ISTAND,NRUSE(73),NRUSE(74)
*       TRAFFIC JAM DETECTOR -- IF A COMPLETE TRAFFIC JAM DETECTED
*       THE REPLICATION AND A FILE WITH THE LOCATION AND
*       STATUS OF EACH VEHICLE IS PRINTED OUT IN FILE 7
*       DETECTION OF A JAM OCCURS IF THE TOTAL NUMBER OF
*       PICKS PROCESSED IN THE CONSOLIDATION AREA IS ZERO
*       FOR 11 CONSECUTIVE OR NONCONSECUTIVE 15 MINUTE INTERVALS
      IF (XX(97).EQ.0) THEN
        XX(110)=XX(110)+1
        IF (XX(110).GT.11) THEN
          GOTO 400
        ENDIF
      ENDIF
      XX(95)=0

```

```

        XX(97)=0
        RETURN
*
*       TROUBLESHOOTING STATS
*
400  WRITE(7,*) , 'TNOW ' , TNOW
      DO 450 J=1, XX(130)
        CALL GETLOC(J, MSGNUM, CPOS, NCPNEX, NCPCUR)
        CALL GETASG(J, NCPDES, IPLOADE, ISTATVU)
        WRITE(7,*) , 'AGV* ' , J, MSGNUM, CPOS, NCPNEX, NCPCUR
        WRITE(7,*) , 'DEST ' , NCPDES, IPLOADE, ISTATVU
450  CONTINUE
      CALL ENTER(50, ATRIB)
      RETURN
      END

      SUBROUTINE ALLOC(I, IFLAG)
      INCLUDE 'PARAM. INC'
      COMMON/SCOM1/ATRIB(MATRB), DD(MEQT), DDL(MEQT), DTNOW, II, MFA,
      1MSTOP, NCLNR, NCRDR, NPRNT, NNRUN, NNSET, NTAPE, SS(MEQT),
      2SSL(MEQT), TNEXT, TNOW, XX(MMXXV)
*
      INTEGER NRANKQ
      REAL A(30), B(30), TEMP
      IFLAG=0
      GOTO (10,30) I
*
      ALLOC 1 HANDLES THE ASRS CRANES -- WHEN AN ENTITY AND
*
      THE RESOURCE THE MODULE ARE REQUESTING ARE BOTH AVAILABLE
*
      (i.e. BOTH ARE AT THE AWAIT NODE), THE MODULE SEIZES THE
*
      RESOURCE (CRANE) AND STARTS INTO THE ACTIVITY OF PICKING
010  J=NNQ(18)
*
      IF AWAIT QUEUE EMPTY, DON'T BOTHER CHECKING
      IF (J.LE.0) THEN
        GOTO 25
      ENDIF
*
      GO THRU ALL THE RESOURCES -- IF A RESOURCE IS AVAILABLE
*
      CHECK ALL THE MODULES IN THE AWAIT QUEUE AND SEE IF
*
      ANYONE WANTS IT.
      DO 15 L=110,169
        IF (NNRSC(L).GT.0) THEN
          TEMP=L
*
          FIND A MODULE THAT WANTS THE RESOURCE
          NRANKQ=NFIND(1,18,7,0,TEMP,0.0)
          IF (NRANKQ.GT.0) THEN
            CALL SEIZE(L,1)
            IFLAG=- (NRANKQ)
            RETURN
          ENDIF
        ENDIF
015  CONTINUE
025  RETURN
*
*
*       ALLOC 2 SENDS PICK MODULES TO THE CONSOLIDATION STANDS
*
      FIRST IN FIRST OUT.

```

```

030 DO 40 J=180,211
      IF (NNRSC(J).GT.0) THEN
        IF (NNQ(20).GT.0) THEN
          CALL SEIZE(J,1)
          CALL COPY(1,20,A)
          A(1)=A(3)
          A(3)=GETARY(3,J-179)
          A(6)=6
          A(8)=J
          CALL FILEM(23,A)
          IFLAG= -1
          RETURN
        ENDIF
      ENDIF
040 CONTINUE
      RETURN
      END

SUBROUTINE OPUT
  INCLUDE 'PARAM.INC'
  COMMON/SCOM1/ATTRIB(MATRB), DD(MEQT), DDL(MEQT), DTNOW, 11, MFA,
  1MSTOP,NCLNR, NCRDR, NPRNT, NNRUN, NNSET, NTAPE, SS(MEQT),
  2SSL(MEQT),TNEXT, TNOW, XX(MMXXV)
  INTEGER IPG(1:3,1:60)
  REAL A(17)
  COMMON/UCOM1/ IPG
  *   FILE 12 OUTPUTS THE NUMBER OF PICKS PROCESSED BY EACH
  *   CONSOLIDATION STAND
  DO 20 I=61,92
    WRITE(12,*) ,XX(I)
  20 CONTINUE
  *   FILE 9 WRITES THE AVERAGE TRIP AND NUMBER OF TRIPS FOR
  *   EACH ROUTE
  DO 40 I=1,13
    AVG=CCAVG(I)
    OBS=CCNUM(I)
    WRITE(9,*) ,AVG,OBS
  40 CONTINUE
      RETURN
      END

  *   THIS SUBROUTINE IS A REPLACEMENT FOR THE ONE IN THE
  *   MHEXI.FOR PROGRAM -- TWO LINES IN THE ORIGINAL WERE COMMENTED
  *   OUT. THEY ARE MARKED BY *****. UNDER CERTAIN CIRCUMSTANCES
  *   THE TWO LINES ALLOWED AN AGV TO KEEP MOVING EVEN THOUGH
  *   IT WAS STOPPED BEHIND A VEHICLE.
  SUBROUTINE VRSETP(IPVU,TOL)
    PARAMETER (KVACP1=1,KVACP2=2,KVALEN=3,KVACAP=4,KVADIR=5,KVANTX=6,
  * KVANCO=7,KVASTO=8,KVAMNO=9,KVATLU=10,KVAPFE=11,KVAPLE=12,
  * KVCSPR=1,KVCNWL=2,KVCPFE=3,KVCPLE=4,KVCMXL=5,KVCPPL=6,
  * KVCCRC=7,KVCNTE=8,KVCCST=9,KVCTSTB=10,KVCTLU=11,KVCVCO=12,
  * KVCRRC=13,KVCCPI=14,KVSESP=1,KVSLSP=2,KVSACC=3,KVSDEC=4,KVSLEN=5,
  * KVSBUF=6,KVSCKZ=7,KVSIFL=8,KVSJRQ=9,KVSINI=10,KVSREP=11,
  * KVSNTL=12,KVSNTU=13,KVSNU1=14,KVSNUU=16,KVSNU2=18,KVSNUF=20,
  * KVSNUC=22,KVSNU3=24,KVSSTE=26,KVSSTF=27,KVSTLU=28,KVSPFE=29,

```

```

* KVSPL=30,KVSPAL=31,KVSNOW=32,KVUPVS=1,KVUCSG=2,KVUCCP=3,
* KVIICP=4,KVUDCP=5,KVUPCL=6,KVUUMO=7,KVUCS1=8,KVUSPD=9,KVUCBF=10,
* KVUCBT=11,KVUTLU=12,KVUSCP=13,KVUSHT=14,KVUNTL=15,KVUSPT=16,
* KVFVSI=4,KVWIFL=4,KVWVSI=5,KVWVRQ=6,KVWVRE=7,
* KVMCP1=8,KVMCP1=4,KVANWF=13,KVCNWF=15,KVFNWF=5,KVMNWF=4,
* KVSNEW=33,KVUNWF=15,KVWNWF=8)
PARAMETER (MXMSG=250)
PARAMETER (MXPOUT=6)
INCLUDE 'PARAM.INC'
COMMON/SCOM1/ATRIB(MATRB), DD(MEQT), DDL(MEQT), DTNOW, 11, MFA,
1MSTOP,NCLNR, NCRDR, NPRNT, NNRUN, NNSSET, NTAPE, SS(MEQT),
2SSL(MEQT),TNEXT, TNOW, XX(MMXV)
COMMON QSET(1)
DIMENSION NSET(1)
EQUIVALENCE (NSET(1),QSET(1))
COMMON/GCOM2/ DTFUL, DDI(MEQT), IICV(MSCND,2), IIDIR(MSCND),
1IIEVT(MSCND), ISEES, LLSCD(MSCND), NNEQD, NNEQS, NNEQT, NNSCD,
2SSI(MEQT), TTOL(MSCND), VVAL(MSCND)
COMMON / VCOM1 / NNEQSU,NNSCDU,VSPEED(MXVUN1)
KSS = NSET(IPVU) + NNEQSU
KSCND = NSET(IPVU) + NNSCDU
IF (IICV(KSCND,2).LE.0) THEN
    SS(KSS) = VVAL(KSCND) + TOL
*****    SSL(KSS) = VVAL(KSCND) + TOL
ELSE
    KSCND0 = IICV(KSCND,2)
    KSS0 = (KSCND0 - NNSCD0) + NNEQSU
    IF (IIDIR(KSCND).GT.0) THEN
        POS = SS(KSS0) - VVAL(KSCND)
    ELSE
        POS = SS(KSS0) + VVAL(KSCND)
    ENDIF
    SS(KSS) = POS + TOL
*****    SSL(KSS) = POS + TOL
ENDIF
RETURN
END

*
* SUBROUTINE EFFIC COMPUTES THE INTERVAL AVERAGE AND
* CUMULATIVE FLEET EFFICIENCY RATES
SUBROUTINE EFFIC
PARAMETER (KVACP1=1,KVACP2=2,KVALEN=3,KVACAP=4,KVADIR=5,KVANTX=6,
* KVANCO=7,KVASTO=8,KVAMNO=9,KVATLU=10,KVAPFE=11,KVAPLE=12,
* KVCSPR=1,KVCNWL=2,KVCPFE=3,KVCPLE=4,KVCMXL=5,KVCPPL=6,
* KVCCRC=7,KVCNTE=8,KVCCST=9,KVCSTB=10,KVCTLU=11,KVCVCO=12,
* KVCRR=13,KVCCP1=14,KVSESP=1,KVSI1SP=2,KVSACC=3,KVSDEC=4,KVSLEN=5,
* KVSBUF=6,KVSCKZ=7,KVSIFL=8,KVSJRQ=9,KVSINI=10,KVSREP=11,
* KVSNTL=12,KVSNTU=13,KVSNU1=14,KVSNUU=16,KVSNU2=18,KVSNUF=20,
* KVSNUC=22,KVSNU5=24,KVSSTE=26,KVSSTF=27,KVSTLU=28,KVSPFE=29,
* KVSPL=30,KVSPAL=31,KVSNOW=32,KVUPVS=1,KVUCSG=2,KVUCCP=3,
* KVIICP=4,KVUDCP=5,KVUPCL=6,KVUUMO=7,KVUCS1=8,KVUSPD=9,KVUCBF=10,
* KVUCBT=11,KVUTLU=12,KVUSCP=13,KVUSHT=14,KVUNTL=15,KVUSPT=16,
* KVFVSI=4,KVWIFL=4,KVWVSI=5,KVWVRQ=6,KVWVRE=7,
* KVMCP1=8,KVMCP1=4,KVANWF=13,KVCNWF=15,KVFNWF=5,KVMNWF=4,

```

```

* KVSNEW 33,KVUNWF-15,KVWNWF-8)
PARAMETER (MXMSG=250)
PARAMETER (MXPOUT 6)
INCLUDE 'PARAM.INC'
COMMON/SCOM1/ ATRIB(MATRB), DD(MEQT), DDL(MEQT), DTNOW, 11, MFA,
1MSTOP,NCINR, NCRDR, NFRNT, NNRUN, NNSET, NTAPE, SS(MEQT),
2SSL(MEQT),TNEXT, TNOW, XX(MMXV)
COMMON / VCOM2 / IPCPBF,IPVUCF,IPVUSG,ISSTOP,KKVCPO(MXVCPO),
*KKVSEG(MXVSEG),KKVSET(MXVSET),KKVUNI(MXVUNI),NNATVU,IPVUBF
CHARACTER*8 CCVCPO(MXVCPO),CCVSET(MXVSET),CCVSEG(MXVSEG)
COMMON / VCOM3 / CCVCPO,CCVSET,CCVSEG
COMMON / VCOM4 / NNVCP0,NNVSEG,NNVSET,NNVUNI
COMMON QSET(1)
DIMENSION NSET(1),QSET(1)
EQUIVALENCE (NSET(1),QSET(1))
IVEH=0
DO 50 I=1,NNVSET
    IPVS=KKVSET(I)
    IF(IPVS.EQ.0) GOTO 50
    CALL AVTREM(IPVS,AVGTLE)
    CALL AVTRFU(IPVS,AVGTLF)
    CALL AVLOAD(IPVS,AVGLOA)
    CALL AVUNLO(IPVS,AVGUNL)
*   TOTAL EFFICIENCY EQUALS TIME TRAVELING TO AN ORDER,
*   TRAVELING TO AN ORDER, LOADING AND UNLOADING.
    TOTPRD=AVGTLE+AVGTLF+AVGUNL+AVGLOA
    XX(123)=XX(123)+1
    IF (XX(123).LE.1) THEN
        XX(124)=TOTPRD
        WRITE(10,*) ,XX(124),TOTPRD
    ELSE
        TOTPRI=(TOTPRD*XX(123))-((XX(123)-1)*(XX(124)))
        WRITE(10,*) ,TOTPRI,TOTPRD
        XX(124)=TOTPRD
    ENDIF
50  CONTINUE
*
*   AVERAGE NUMBER OF MODULES/PALLETS WAITING FOR AGVs
*   COLLECT THE AVERAGE NUMBER OF MODULES WAITING FOR AGV
*   THE VWAIT STATISTICS ARE POLLED TO GET THE STATISTICS
    XX(121)=XX(121)+1
    AVQCUM=FFAVG(13)+FFAVG(16)+FFAVG(19)+FFAVG(21)
    AVQCUM=AVQCUM+FFAVG(24)+FFAVG(25)+FFAVG(26)+FFAVG(29)
    AVQCUM=AVQCUM+FFAVG(30)
    IF (XX(121).LE.1) THEN
        XX(122)=AVQCUM
        WRITE(10,*) ,XX(122)
    ELSE
        AVQINT=(AVQCUM*XX(121))-((XX(121)-1)*XX(122))
        WRITE(10,*) ,AVQINT
        XX(122)=AVQCUM
    ENDIF
RETURN
END

```

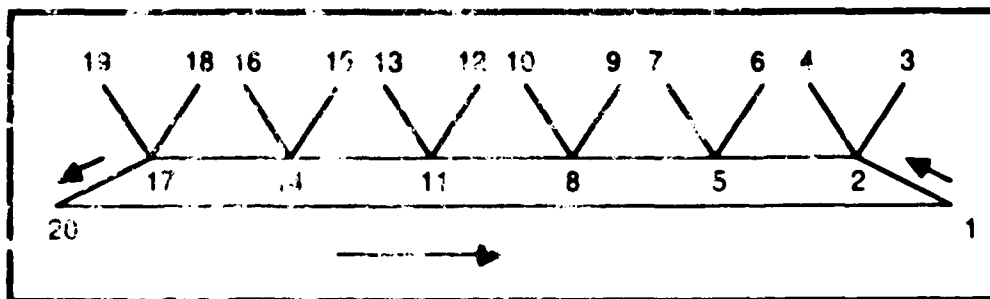
### Appendix C. Parameter File

Use this code to redimension SLAM. The code is also used when the fortran code is compiled.

```
C PARAM.INC
C MATERIAL HANDLING EXTENSION VERSION 2.1 PARAMETERS
C
  PARAMETER (MXCRN=20, MXSPA=75, MXPIL=200, MXTRK=10,
1           MXCPK=10, MXYLOC=21000, MXVCPO=650,
2           MXVSEG=700, MXVSET=10, MXVUNI=50)
C
C SLAM 11 REGULAR VERSION PARAMETERS
C
  PARAMETER (MEQT=100, MSCND=35, MENTR=60, MRSC=260, MARR=15,
1  MGAT=25, MHIST=50, MCELS=500, MCLCT=50, MSTAT=150, MEQV=100,
2  MATRB=100, MFILS=100, MPLOT=10, MVARP=10, MSTRM=10,
3  MACT=100, MNODE=500, MITYP=50, MMXXV=150, MMXFLD=125)
  PARAMETER ( MAXLVL=50, MXMACS=20, MXBRKS=10)
C
  PARAMETER (MVARP1=MVARP+1)
```

## Appendix D

Appendix D contains the figures needed to decode the VCPOINT and VSGMENT statements in the simulation program in appendix A. Each figure is a certain area of the IMC with the control points marked. The figures are not drawn to scale. To use the figures to find a segment, look for the starting or ending control points.



**Figure 36. Control Points - Module Loading Area**

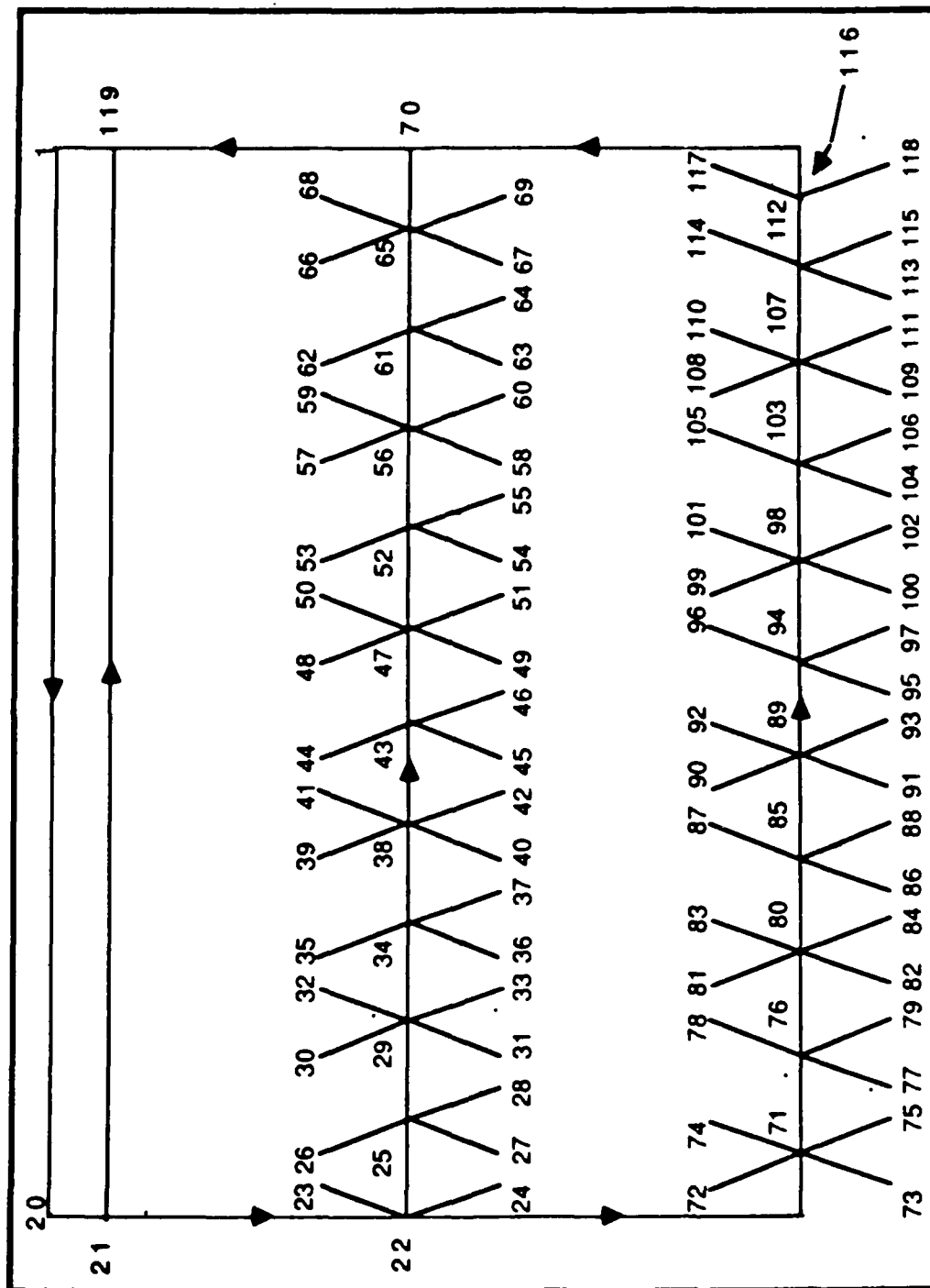


Figure 37. Control Points - Module Storage Area



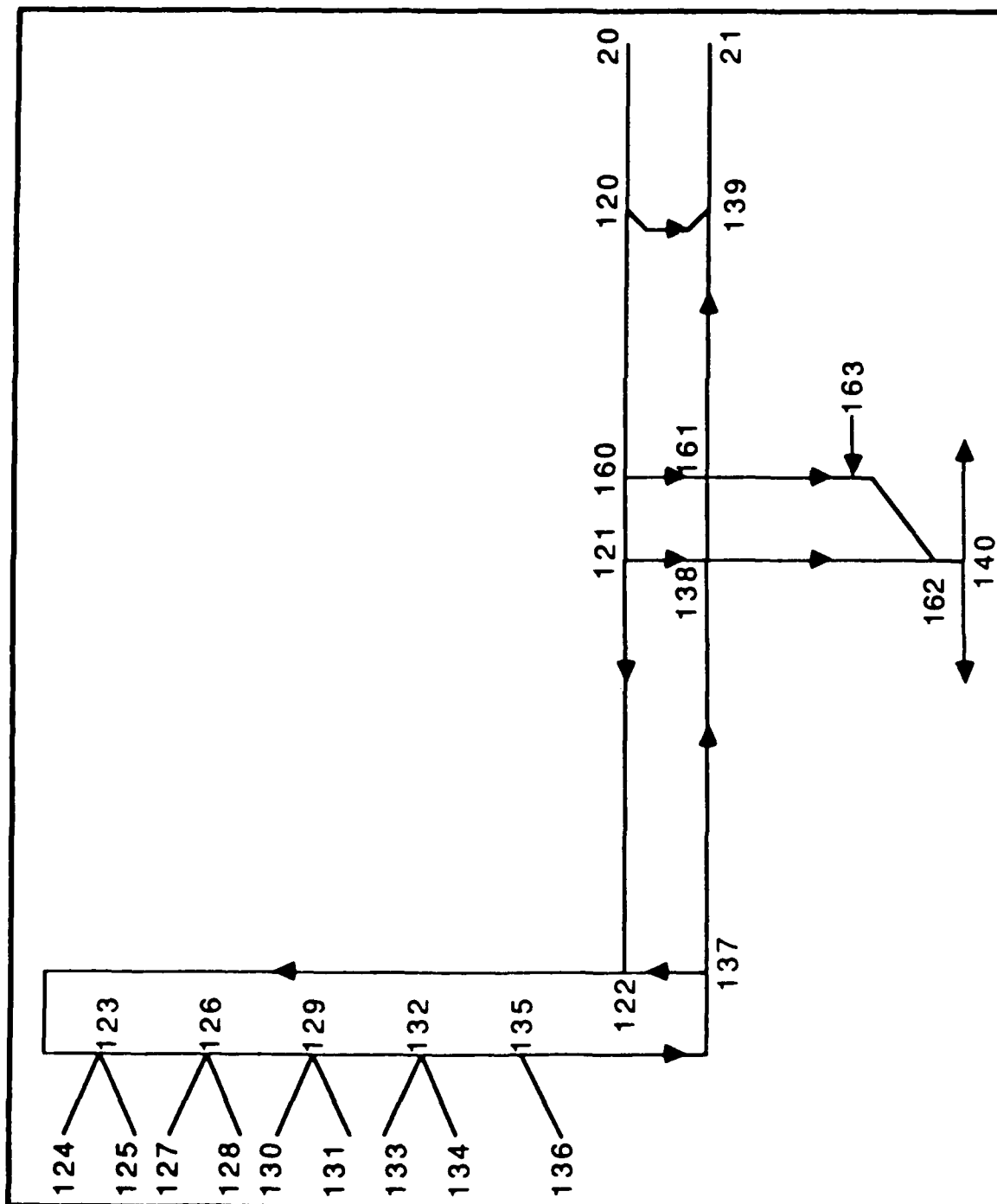


Figure 38. Control Points - 3 P&M

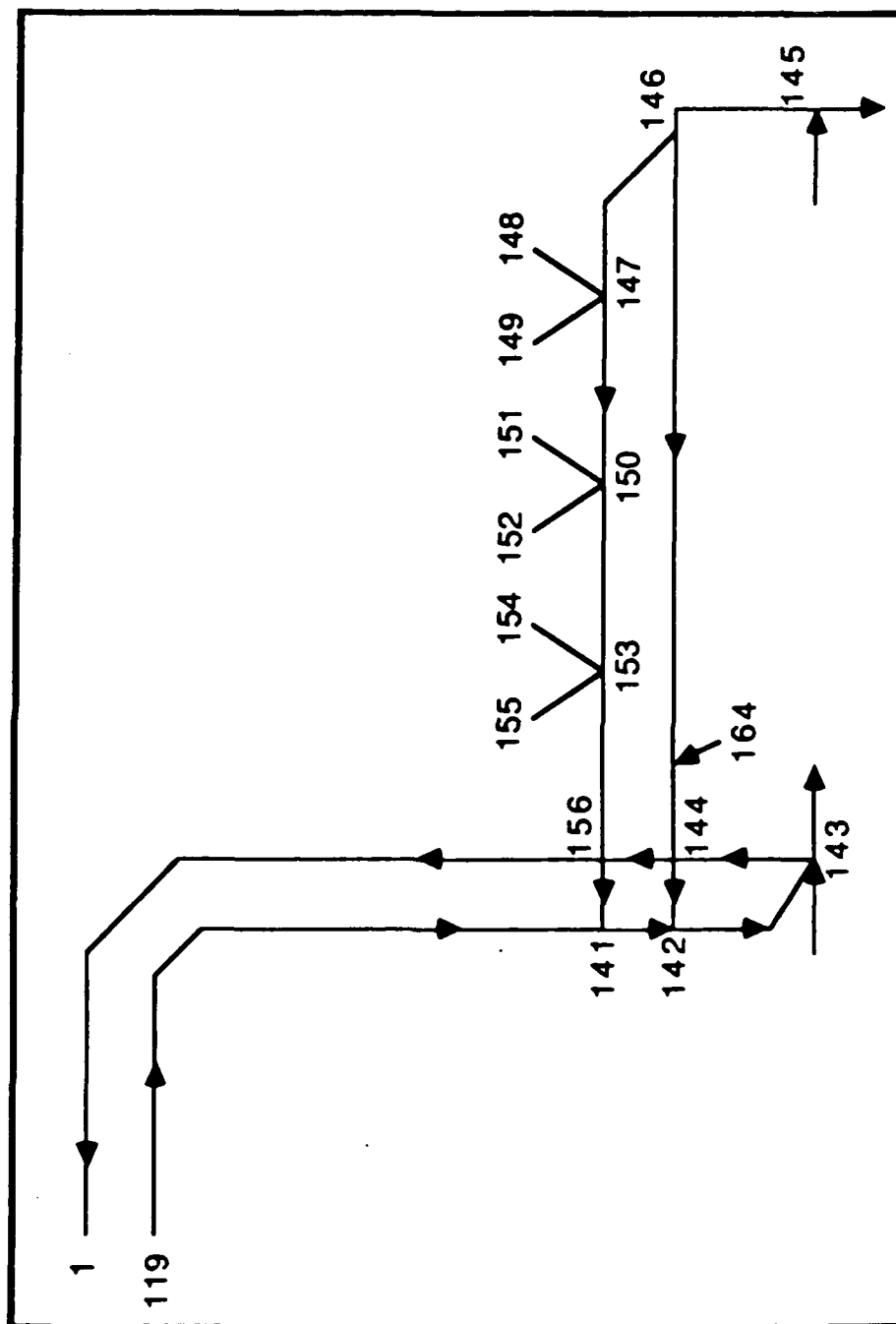
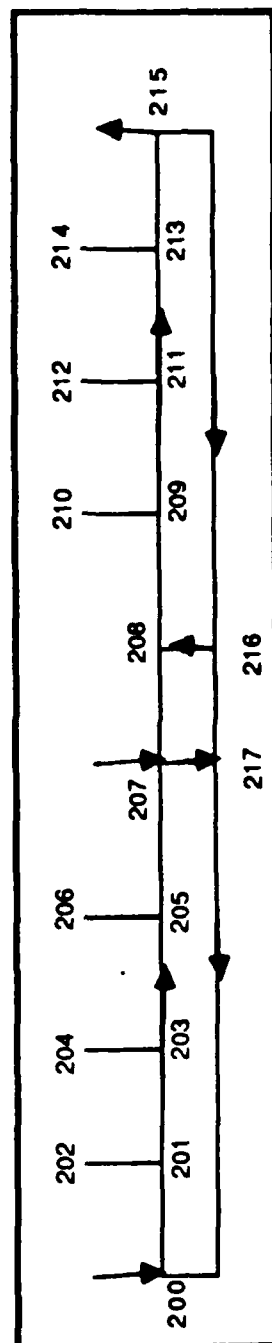
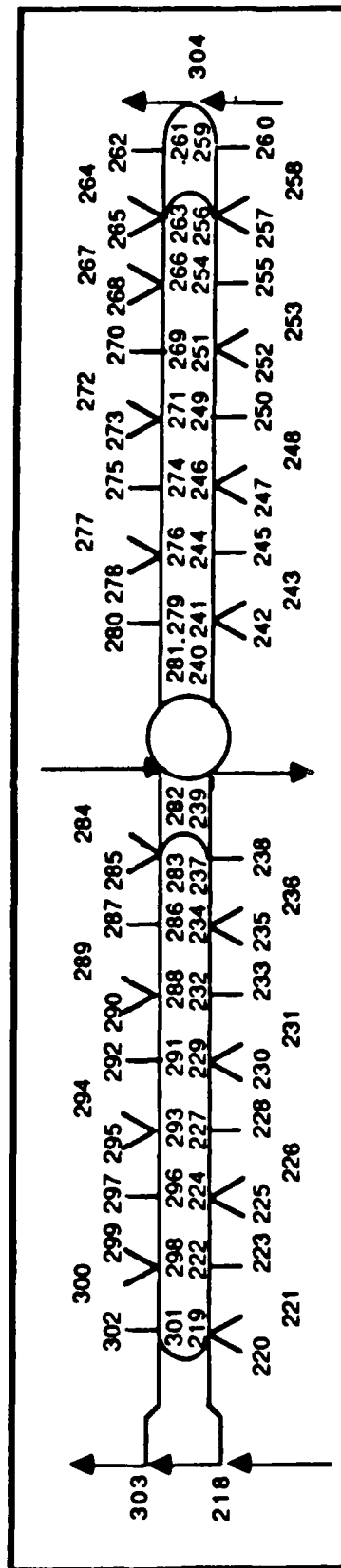


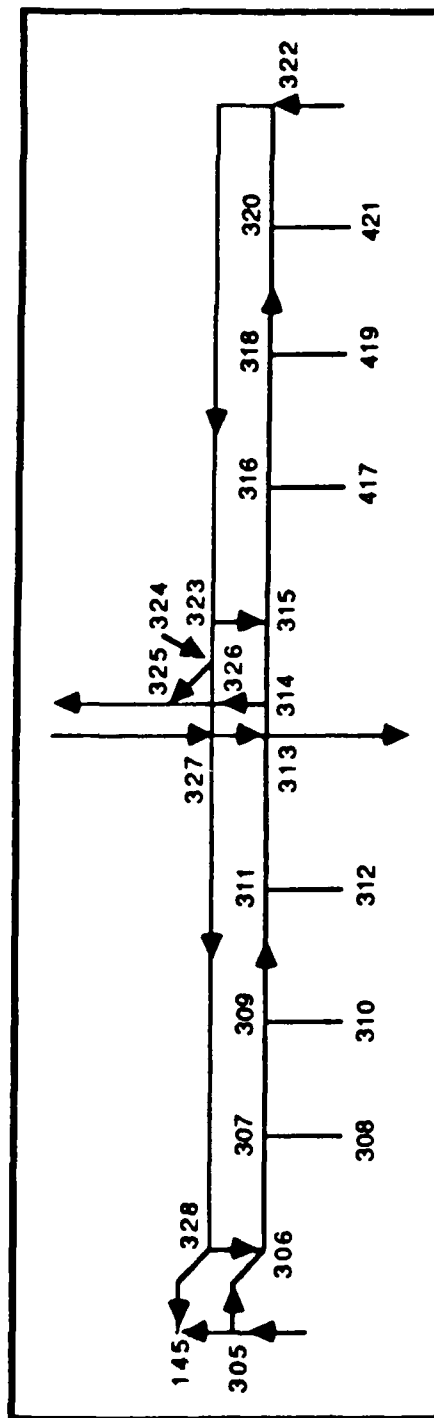
Figure 39. Control Points - LTL



**Figure 40. Control Points - South End Of Highrise**

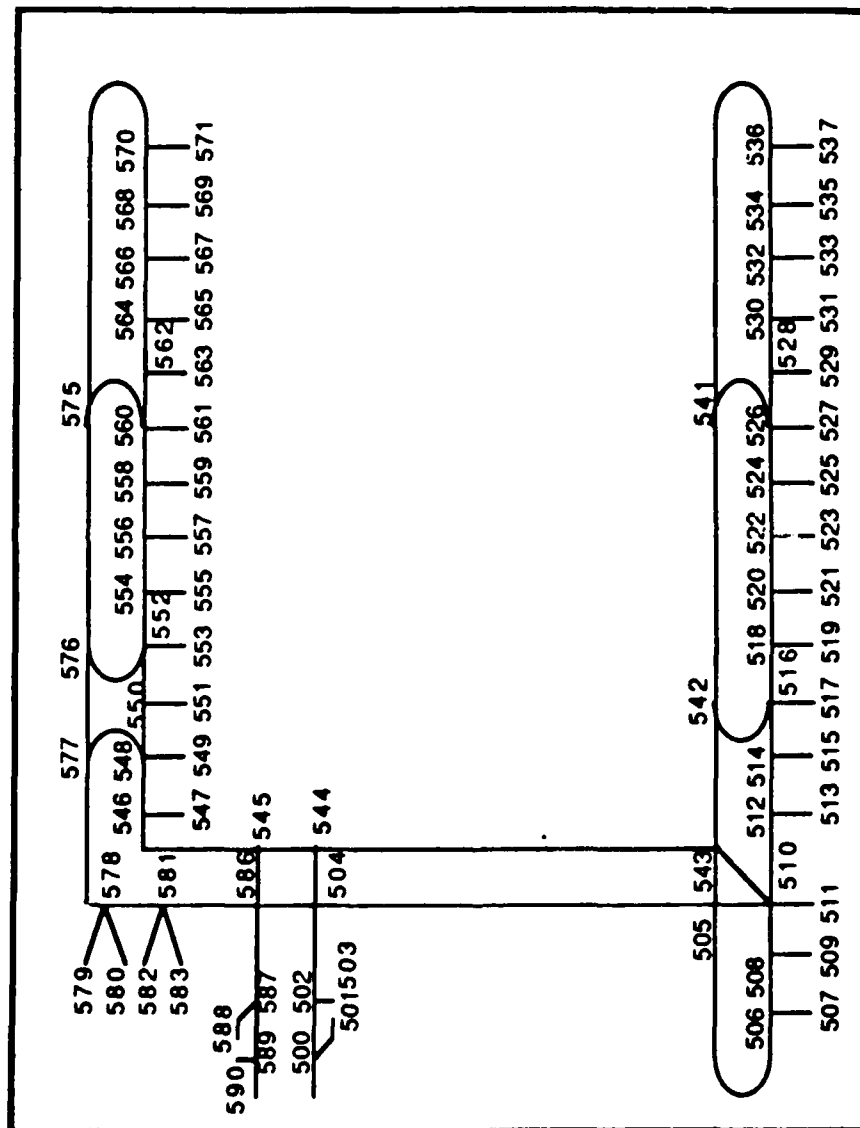


**Figure 41. Control Points - Middle Highrise**



**Figure 42. Control Points - North End Of Highrise**





**Figure 44. Control Points - Pallet Storage**

## Bibliography

1. Arthur, Jeffrey L. "Microcomputer Simulation Systems," *Computers and Operations Research*, 13: 167-183 (No. 2/3 1986).
2. Ashayeri, J. and L. F. Gelders. "Warehouse Design Optimization," *European Journal of Operational Research*, 21: 285-294 (September 1985).
3. AutoSimulations, Incorporated. *Project Final Report for the Defense Logistic Agency Integrated Material Complex Mechanized Material handling System*. Contract DLA410-86-C-006. 31 July 1986.
4. Azadivar, Farhad. "A Simulation Optimization Approach To Optimum Storage And Retrieval Policies in an Automated Warehousing System," *Proceedings of the 1984 Winter Simulation Conference*. 207-214. New York: IEEE Press, 1984.
5. Cooley, Belva J. and John W. Cooley. "Data Analysis For Simulation Experiments: Application of a Distribution-Free Multiple Comparisons Procedure," *Decision Sciences*, 11: 483-493 (1980).
6. Davis, Deborah A. "Modeling AGV Systems," *Proceedings of the 1986 Winter Simulation Conference*. 568-574. New York: IEEE Press, 1986.
7. Fitzgerald, Kevin R. "How to Estimate the Number of AGVs You Need," *Modern Materials Handling*, 4 : 79 (October 1985).
8. Grant, Floyd H. and James R. Wilson. "Material Flow Evaluation in Robotic Systems Using Simulation," *Robotics and Material Flow*, edited by S. Y. Nof. Amsterdam: Elsevier, 1986.
9. Han, Min-Hong and others. *On Sequencing Retrievals in an Automated Storage/Retrieval System*. Atlanta: Georgia Institute of Technology, October 1984 (AD-A150 613).
10. Hoover, Stewart V. and Ronald F. Perry. "Validation of Simulation Models: The Weak/Missing Link," *Proceedings of the 1984 Winter Simulation Conference*. 293-296. New York: IEEE Press, 1984.
11. Koff, Garry A. "Driverless Trains and Wire-Guided Pallet Trucks," *Materials Handling Handbook* (Second Edition), edited by Raymond A. Kulwiec. New York: John Wiley and Sons, 1985.



12. Law, Averill M. and Michael G. McComas. "Pitfalls in Simulation of Manufacturing Systems," *Proceedings of the 1986 Winter Simulation Conference*. 539-543. New York: IEEE Press, 1986.
13. Maxwell, W. L. and J. A. Muckstadt. "Design of Automatic Guided Vehicle Systems," *IIE Transactions*, 14: 114-125 (June 1982).
14. Phillips, Don T. "Simulation of Material Handling Systems: When and Which Methodology?," *Industrial Engineering*, 12: 65-77 (September 1980).
15. Pritsker, A. Alan B. *Introduction to Simulation and SLAM II*. New York: Halsted Press, 1986.

### Vita

Captain Richard B. Berry was born on 29 August 1959 in Inglewood, California. He graduated from Meadowdale High School in Lynnwood, Washington in June of 1977. After graduation he attended the University of Washington, from which he received a Bachelor of Arts in Mathematics in March of 1982. Three months later, he received a commission as a Second Lieutenant in the United States Air Force through the Reserve Officer Training Corp, Detachment 910. Called to active duty in October 1982, he reported to the 6585 Test Group at Holloman AFB, New Mexico. There, he worked as a Numerical Methods Analyst. In October 1985, he became the Executive Officer to the Test Group Commander, until entering the School of Engineering, Air Force Institute of Technology, in May of 1986.

Permanent Address: 18921 84th W.  
Edmonds, Washington 98020

## REPORT DOCUMENTATION PAGE

Form Approved  
OMB No. 0704-0188

1a REPORT SECURITY CLASSIFICATION UNCLASSIFIED			1b RESTRICTIVE MARKINGS		
2a SECURITY CLASSIFICATION AUTHORITY			3 DISTRIBUTION / AVAILABILITY OF REPORT		
2b DECLASSIFICATION / DOWNGRADING SCHEDULE					
4 PERFORMING ORGANIZATION REPORT NUMBER(S) AFIT/GOR/ENS/87D-2			5 MONITORING ORGANIZATION REPORT NUMBER(S) Approved for public release; distribution unlimited.		
6a NAME OF PERFORMING ORGANIZATION School of Engineering	6b OFFICE SYMBOL (If applicable) AFIT/ENS		7a NAME OF MONITORING ORGANIZATION		
6c ADDRESS (City, State, and ZIP Code) Air Force Institute of Technology Wright-Patterson AFB, OH 45433-6583			7b ADDRESS (City, State, and ZIP Code)		
8a NAME OF FUNDING / SPONSORING ORGANIZATION Defense Logistics Agency	8b OFFICE SYMBOL (If applicable) DLA/DORO		9 PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER		
8c ADDRESS (City, State, and ZIP Code) DLA/DORO Richmond, VA 23297-5000			10 SOURCE OF FUNDING NUMBERS		
			PROGRAM ELEMENT NO	PROJECT NO	TASK NO
			WORK UNIT ACCESSION NO		
11 TITLE (Include Security Classification) See box 19					
12 PERSONAL AUTHOR(S) Richard B. Berry, B.A., Capt, USAF					
13a TYPE OF REPORT MS Thesis	13b TIME COVERED FROM Mar 87 to Dec 87		14 DATE OF REPORT (Year, Month, Day) 1987 December		15 PAGE COUNT 173
16 SUPPLEMENTARY NOTATION					
17 COSATI CODES			18 SUBJECT TERMS (Continue on reverse if necessary and identify by block number)		
FIELD	GROUP	SUB-GROUP	Materials Handling Equipment, Automated Guided Vehicles (AGVs) Warehouse Automation, Simulation, Automated Storage and Retrieval Machines		
13	09				
13	06				
19 ABSTRACT (Continue on reverse if necessary and identify by block number)					
Title: A Simulation of the Defense Logistics Agency's Integrated Materials Complex					
Thesis Chairman: Joeseeph R. Litko, Major, USAF Associate Professor of Operations Research					
20 DISTRIBUTION / AVAILABILITY OF ABSTRACT <input type="checkbox"/> UNCLASSIFIED/UNLIMITED <input checked="" type="checkbox"/> SAME AS RPT <input type="checkbox"/> DTIC USERS			21 ABSTRACT SECURITY CLASSIFICATION Unclassified		
22a NAME OF RESPONSIBLE INDIVIDUAL Joeseeph R. Litko, Major, USAF			22b TELEPHONE (Include Area Code) 1-513255-3362		22c OFFICE SYMBOL AFIT/ENS

Approved for Release by NSA on 09-11-2013 pursuant to E.O. 13526  
 24 Feb 88

The purpose of this research was to show the effects of the number of automated guided vehicles (AGVs), the number of pickup and delivery stands and the storage and retrieval machines have on the Defense Logistics Agency's Integrated Materials Complex. A computer simulation model was designed using the SLAM simulation language with the materials handling extensions.

Four different AGV fleet sizes and two automated storage and retrieval machine picking modes were simulated. The performance measure of the IMC is to take 15,500 items from the highrise rack and bin storage area, and process for shipping, all in an eight hour shift.

The results showed that the performance goal was met or exceeded, regardless of the AGV fleet size (20 to 35 vehicles) or the pick mode used. Analysis of the data confirmed that the AGV fleet size and pick mode had no significant effect on achieving the performance goal.

However, if the performance goal was increased, or a smaller vehicle fleet used, the AGV fleet size would start to have an effect on the performance. A smaller vehicle fleet operating at a high efficiency, would be hard pressed to handle any additional work, if the performance goal was increased. A higher performance goal may also cause the pick mode to have more of an effect on the operation of the IMC.

END  
DATE  
FILMED  
MARCH  
1988  
DTIC